SAP liveCache 7.5 and MaxDB 7.5 on IBM TotalStorage

DS 8000 / ESS / SVC

Integration of the SAP SCM liveCache hot standby storage in an AIX environment

RTEHSS & IBM TotalStorage® integration

Oliver Goos
Carol Davis / Herbert Diether / Werner Thesing

Note: Before using this information and the product it supports, read the information in ”Notices and Disclaimers” on page 65.

Version 3.0

Table of Content

Figures

Introduction

In an SAP SCM environment, liveCache is becoming an increasingly mission critical component. liveCache, like any other database technology, can be provided with a fail-over solution using traditional cluster implementations. An HACMP fail-over solution for liveCache has been available on AIX, and endorsed by SAP, for several years (starting with LC version 7.2). A traditional fail-over solution, where the database disks are taken over by the backup server and the database recovered and restarted, has several disadvantages in a large liveCache implementation. The benefits and performance advantages of liveCache result from its ability to build a very large memory cache and perform specially tailored functions against these in-memory data structures.

The memory cache for large implementations may exceed many gigabytes in size. A traditional fail-over solution will need to restart the database and rebuild this memory cache. This represents a large time delay, before liveCache is ready to resume production. Before this rebuild activity can even begin, a fail-over solution must acquire and activate the disks of the failed system and perform database recovery. All of these activities will increase with the size of the liveCache, both on disk and in memory. The larger the liveCache, the greater its importance is likely to be in the SCM landscape; and the longer the expected fail-over and recovery time.

In order to provide the fastest possible means of recovering, SAP has introduced new powerful hot-standby functionality with liveCache 7.5, available with SCM 4.1. While this functionality has been implemented for both the MaxDB and liveCache, this document will focus on the liveCache for SCM. The objective of this implementation is to provide the following characteristics:

* speed of recovery and return to production
* coverage of server outage
* coverage of database outage
* coverage of data disk failures
* automated failover and fallback
* designed for no performance impact to the master
* ease of management capability for DB administrators

The solution design for the liveCache hot-standby requires specific functionality on behalf of the supporting I/O subsystem (split mirror and concurrent volume access) and is closely integrated with the control software of the subsystem via an API. The integration of this solution requires an I/O subsystem specific shared library, mapping the SAP requirements to the subsystem, and a cluster solution on the part of the server platform to manage the fail-over control and IP access. IBM offers this solution for TotalStorage™ on AIX for

- SAN Volume Controller (and supported I/O subsystems);
- Enterprise Storage Server;
- DS8000.

HACMP is used as high availability software and provides the cluster functionality on AIX.

This document describes in detail the implementation of this solution running on pSeries server with AIX 5.3 attached to the IBM TotalStorage DS8000 and using HACMP 5.x as high availability software. It will also include information for the ESS and SVC implementation.

Executive solution summary

Product name and description

- MaxDB hot standby storage with IBM TotalStorage DS 8000, ESS and SVC.
- SAP liveCache hot standby storage with IBM TotalStorage DS 8000, ESS and SVC.
- Provides one or more hot standby SAP liveCache / MaxDB server for an application to fail over in a minimum amount of time.
- Database volumes (log and data) could be hosted on one storage systems, using ‘flash copy’ function.
- A services offering through IBM Storage Services.

Target Customer and Supported Platform

- customer with SAP SCM liveCache or MaxDB
- IT business continuity requires a tier 7 solution
- running AIX 5L.

Customer Need and Benefit

- SAP customers running SAP SCM APO or MaxDB application will have their interim data backup and protected.
- In the event of a server outage, the APO application will have the most recent cache data to use based on user set frequency.

Outlook

- Other IBM TotalStorage products could be implemented by request.
- Support of two Storage subsystems, using remote copy functions, could be implemented by request.

Preface

The aim of this document is to describe how the hot standby solution for MaxDB 7.5 and SAP liveCache could be implemented with the IBM TotalStorage products DS 8000 (IBM 2107); Enterprise Storage Server (IBM2107) and SAN Volume Controller (IBM2145).

This paper will guide you through the volume set up on the host system, enabling the hot standby functionality for the MaxDB / SAP liveCache, installing the storage dependent software library, tailoring the configuration file and installing the database. Also it is briefly described how to enable the hot standby solution and how to do basic tasks (e.g. start, stop …). The tailoring of the high availability software (HACMP) is explained. All thinks considered and after reading this whitepaper, you will be able to implement the 'hot standby storage' for SAP liveCache and the MaxDB.

This solution is based on an earlier project which was performed to support the liveCache Hot Standby Storage (HSS) on the ESS and the SVC. The former project was running at

- European Storage Competence Center (ESCC) in Mainz / Germany
- IBM SAP International Competence Center (ISICC) in Walldorf / Germany.

Acknowledgments

It is obvious that an integration solution like this cannot be run by a single person. Many people have helped to make this project a success and special thanks is given to all people that participated in one or the other way. These persons are:

The group that was working on the solution:

Oliver Goos	Storage specialist / developer	IBM Germany
Carol Davis	Senior pSeries / AIX Specialist	IBM Germany
Werner Thesing	liveCache/MaxDB support	SAP AG Germany
Herbert Diether	SAP R/3 on pSeries/HACMP	SERCON Germany

The sponsors, which helped with manpower and budget:

Eva Lau	Business Development & Alliances	IBM USA
Keith Murray	Manager ISSIC Walldorf	IBM Germany
Jörg Hoffmeister	Development Manager NW DT MaxDB&liveCache	SAP AG Germany
Peter Cronauer	Manager TotalStorage ATS Services	IBM Germany

For technical support:

Jens Wissenbach	Test Programmer	IBM Germany
Jörg Mensing	liveCache/MaxDB developer	SAP AG Germany

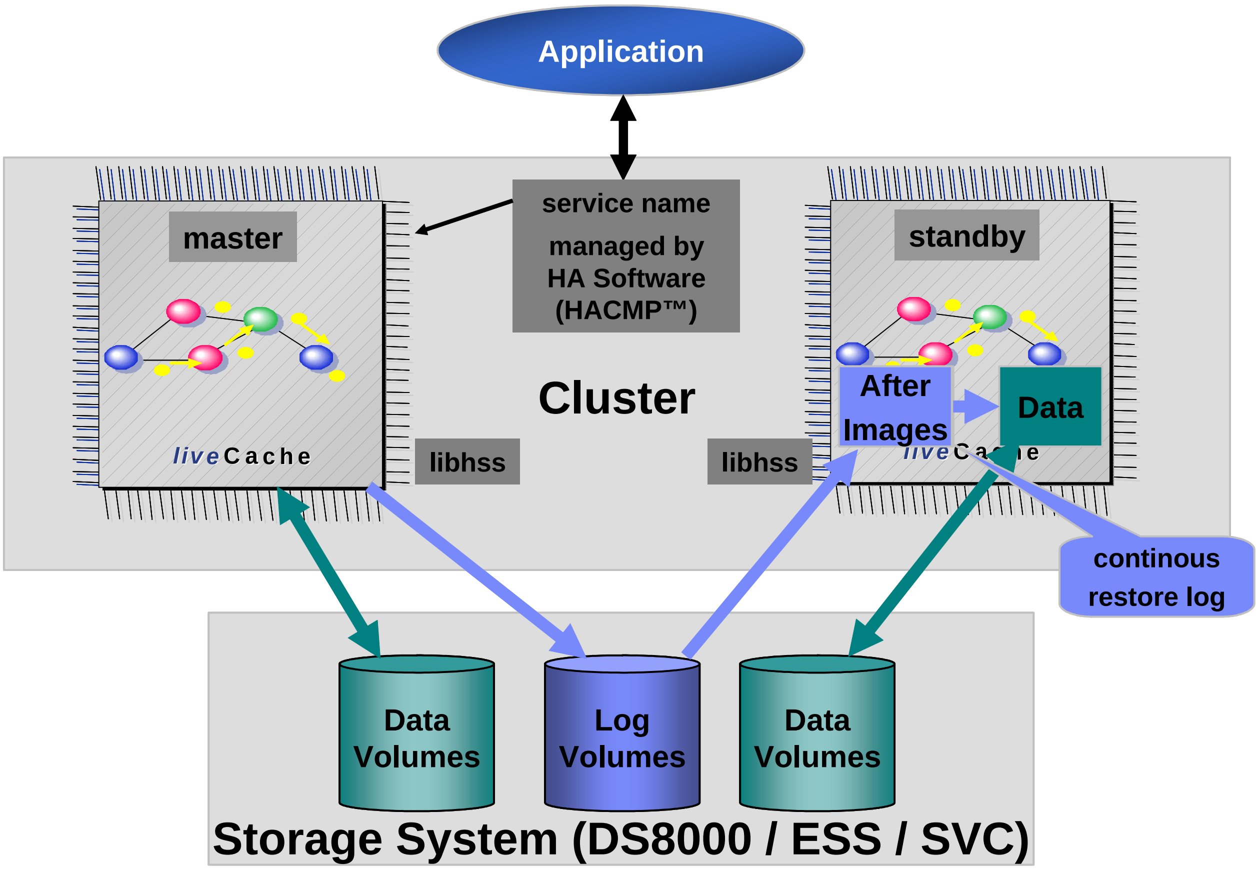
Edition History

Version 1.0	first draft	(Oliver Goos, IBM, Mainz)
Version 1.1	Database related part added	(Werner Thesing, SAP AG, Walldorf)
Version 1.2	HACMP part added	(Herbert Diether, ISICC, Walldorf)
Version 1.3	Introduction added	(Carol Davis, ISICC, Walldorf)
Version 1.4	DBMGUI part added	(Werner Thesing, SAP AG, Walldorf)
Version 1.9	pre final release	(Oliver Goos, IBM, Mainz)
Version 2.0	final release	(Oliver Goos, IBM, Mainz)
Version 3.0	add DS8000 implementation	
	reflect changes in ESScli	(Oliver Goos, IBM, Mainz)

1 Overview

The Base of the Hot Standby Storage is a fail over system, consisting of several (at least two) physically separated database server with a storage system that is physically shared between the instances. The controlling instance of the fail over system is used to detect the fail over situation and perform the operations needed to redirect client connections. The HotStandby implementation is based on two or more separated database servers that access a single storage system.

Each database server must have an own unique network address. The network connection switch is not part of the system and must be solved by third party software. The data volumes are separated, but the log volume is shared.



2 Requirements

2.1 Storage systems

- IBM TotalStorage Enterprise Storage Server (ESS)
- IBM TotalStorage SAN Volume Controller (SVC)
- IBM TotalStorage DS 8000

2.2 Operating systems

- AIX 5L

2.3 Network

- TCP/IP connections from each HSS node to the supported storage system.

2.4 Host systems

- IBM P-series

2.5 Database

- MaxDB or SAP liveCache Version > 7.5.0

2.6 Other

- HighAvailableClusterMultiprocessing (HACMP) Version 4.5 or higher
- storage dependent software library
- IBM 2105 (ESS)
  ESScli / IBM2105CLI for all HSS nodes
  enhanced copy function: FlashCopy V2
- IBM 2145 (SVC)
  IBM2145CLI for all HSS nodes
  SecureShell (SSH) is required on each HSS Node
- IBM 2107 (DS8000)
  DScli on all HSS nodes
  enhanced copy function: FlashCopy

3 Setting up the hosts

This chapter gives an overview how to set up the volume access for the nodes of the cluster. It will not explain how to install the OS nor how to connect the different components with the networks and how to set up the switches.

Both hosts are running with AIX 5L and have a connection to the TCP/IP network and a connection to the Storage Area Network (SAN) via Fibre Channel (FC). Within the SAN you need an ESS / SVC / DS8000, which will host the volumes for the database.

Create **n** log volumes and 2**m** data volumes on the storage system (**n ε N, m ε N**). Since the log volumes will be accessed from both hosts, they must be assigned to both. The group of data volumes will be split into two parts, each volume from group **A** needs a corresponding volume of the identical size in group **B**.

To support multipathing the decision must be made which way to choose.

1. The Subsystem Device Driver (SDD)
   is a pseudo device driver designed to support the multipath configuration environments in the IBM TotalStorage Enterprise Storage Server, the IBM TotalStorage DS family, the IBM TotalStorage SAN Volume Controller, and the IBM TotalStorage SAN Volume Controller for Cisco MDS 9000.
2. The Subsystem Device Driver Path Control Module (SDDPCM)
   is a loadable path control module designed to support the multipath configuration environment in the IBM TotalStorage Enterprise Storage Server and the IBM TotalStorage DS family.

Download the relevant packages from http://www.storage.ibm.com/ and install them on both hosts. The following documentation assumes the usage of the SDDPCM on AIX 5.3. Therefore the packages `devices.fcp.disk.ibm.mpio.rte` and `devices.sddpcm.53.rte` where downloaded from the WEB site and applied.

For detailed information refer to the appropriate **Host Attachment Guide**.

Assuming that the database will use raw devices, e.g. raw logical volumes, we need to create the volume groups and logical volumes on the nodes after we created the volumes on the storage system.

3.1 HSS_NODE_001 => MASTER

Two steps are required to set up the master node of the cluster. Create the appropriate volumes and setting up the database. Next chapter will describe how to set up the volumes using smitty and the command line. For the database I used the example script which comes with the MAX DB 7.5 package as a base and modified it to fit the requirements. More information about creating a database instance could be found in chapter **3.1.3 Create the database** on page **14**.

3.1.1 Set up the volumes

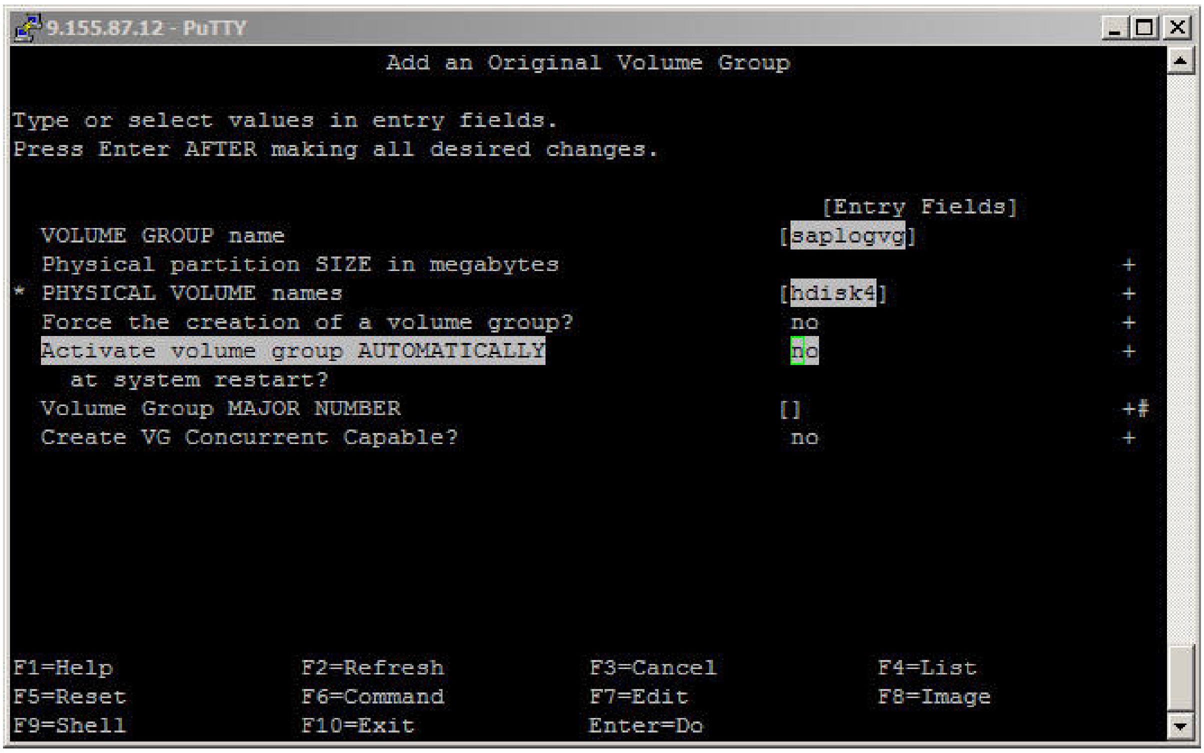
After all volumes are created on the storage system, create the volume groups and logical volumes on the nodes. First of all gather the information about the volumes on the DS. Therefore use the command **`pcmpath query device`**. In Figure 1 on page 6 an example of the setup could be found.

```
# pcmpath query device

DEV#:   4  DEVICE NAME: hdisk4  TYPE: 2107900  ALGORITHM:  Load Balance
SERIAL: 75730411400
==========================================================================
Path#      Adapter/Path Name          State     Mode     Select     Errors
    0           fscsi0/path0          CLOSE   NORMAL          0          0
    1           fscsi0/path1          CLOSE   NORMAL          0          0
    2           fscsi1/path2          CLOSE   NORMAL          0          0
    3           fscsi1/path3          CLOSE   NORMAL          0          0

DEV#:   5  DEVICE NAME: hdisk5  TYPE: 2107900  ALGORITHM:  Load Balance
SERIAL: 75730411401
==========================================================================
Path#      Adapter/Path Name          State     Mode     Select     Errors
    0           fscsi0/path0          CLOSE   NORMAL          0          0
    1           fscsi0/path1          CLOSE   NORMAL          0          0
    2           fscsi1/path2          CLOSE   NORMAL          0          0
    3           fscsi1/path3          CLOSE   NORMAL          0          0
```

Figure 1: example for ‘pcmpath query device’

In this case, the `hdisk4` is used as log volume and `hdisk5` as data volume. To create the volume group, use the fast path ‘`smitty _mkovg`’, or the command line. Create both volume groups with the option ‘Activate volume group AUTOMATICALLY at system restart?’ set to no.

Example with ‘smitty _mkovg’:



```
                      Add an Original Volume Group

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
  VOLUME GROUP name                                 [saplogvg]
  Physical partition SIZE in megabytes                                      +
* PHYSICAL VOLUME names                             [hdisk4]                +
  Force the creation of a volume group?              no                     +
  Activate volume group AUTOMATICALLY                no                     +
    at system restart?
  Volume Group MAJOR NUMBER                         []                     +#
  Create VG Concurrent Capable?                      no                     +

F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 2: fastpath 'smitty _mkovg'

Or use the command line to create the volume groups.

In this example:

```
# /usr/sbin/mkvg -y saplogvg -n hdisk4
# /usr/sbin/mkvg -y sapdatavg -n hdisk5
```

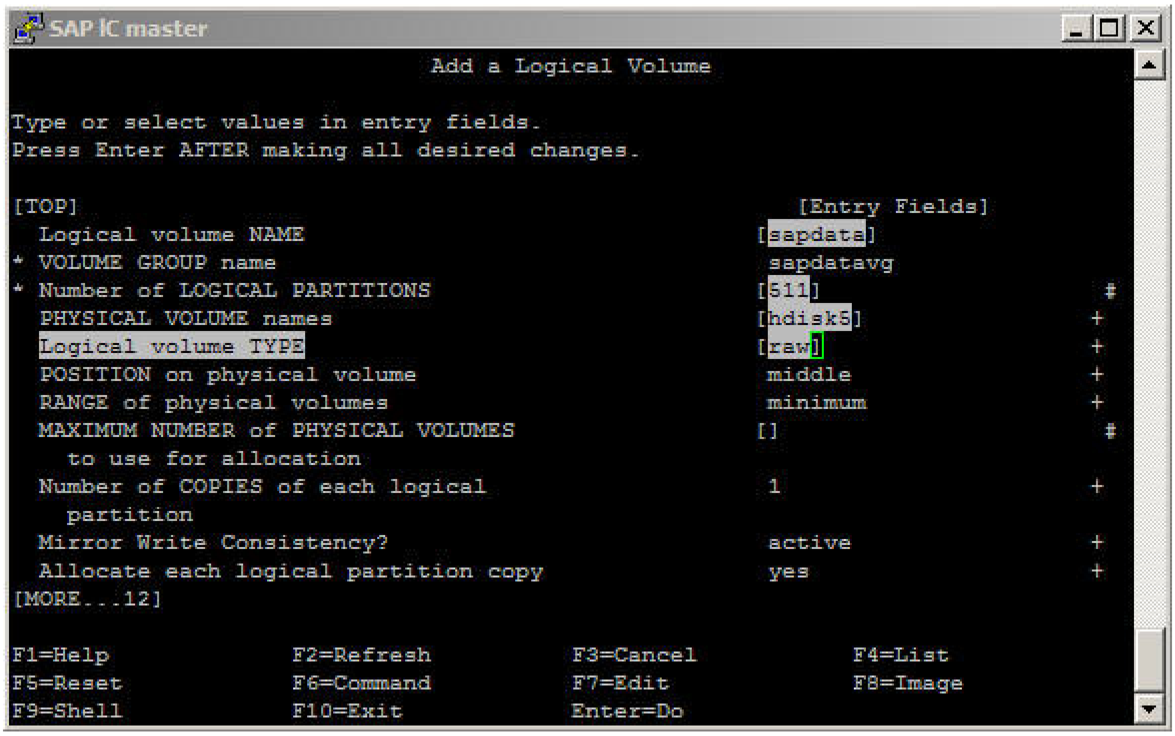
Check the size of the new volume groups with

```
# lsvg sapdatavg
VOLUME GROUP:       sapdatavg                VG IDENTIFIER:  00cd74fe00004c00000001052bb3d7ec
VG STATE:           active                   PP SIZE:        8 megabyte(s)
VG PERMISSION:      read/write               TOTAL PPs:      511 (4088 megabytes)
MAX LVs:            256                      FREE PPs:       511 (4088 megabytes)
LVs:                0                        USED PPs:       0 (0 megabytes)
OPEN LVs:           0                        QUORUM:         2
TOTAL PVs:          1                        VG DESCRIPTORS: 2
STALE PVs:          0                        STALE PPs:      0
ACTIVE PVs:         1                        AUTO ON:        no
MAX PPs per VG:     32512                                    0
MAX PPs per PV:     1016                     MAX PVs:        32
LTG size (Dynamic): 256 kilobyte(s)          AUTO SYNC:      no
HOT SPARE:          no                       BB POLICY:      relocatable
```

Now set up the logical volumes.

Call ‘smitty mklv’:

Choose the appropriate volume group, fill in the required fields and press **ENTER**.

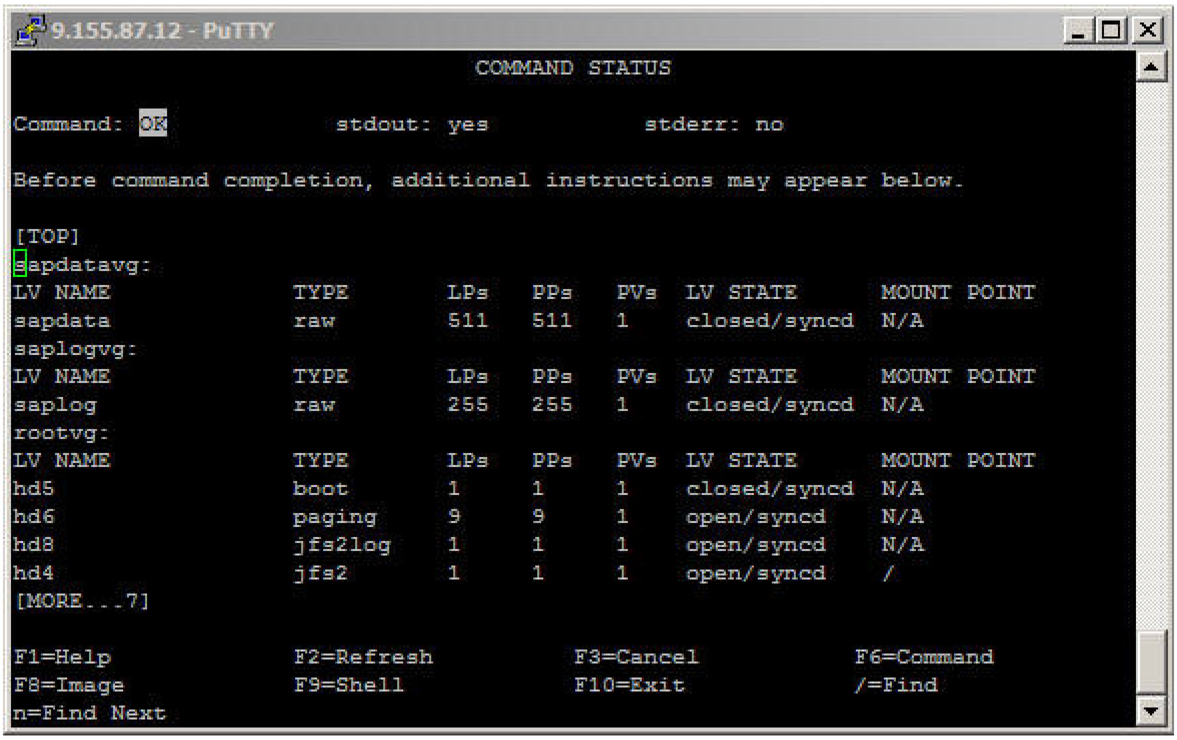


```
                         Add a Logical Volume

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

[TOP]                                                  [Entry Fields]
  Logical volume NAME                                 [sapdata]
* VOLUME GROUP name                                    sapdatavg
* Number of LOGICAL PARTITIONS                        [511]             #
  PHYSICAL VOLUME names                               [hdisk5]         +
  Logical volume TYPE                                 [raw]            +
  POSITION on physical volume                          middle          +
  RANGE of physical volumes                            minimum         +
  MAXIMUM NUMBER of PHYSICAL VOLUMES                  []                #
    to use for allocation
  Number of COPIES of each logical                     1               +
    partition
  Mirror Write Consistency?                            active          +
  Allocate each logical partition copy                 yes             +
[MORE...12]

F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 3: smitty create raw logical volume

Example for the command line:

```
# mklv –y saplog –t raw saplogvg 255 hdisk4
# mklv –y sapdata –t raw sapdatavg 511 hdisk5
```



```
                              COMMAND STATUS

Command: OK            stdout: yes           stderr: no

Before command completion, additional instructions may appear below.

[TOP]
sapdatavg:
LV NAME             TYPE       LPs   PPs   PVs  LV STATE      MOUNT POINT
sapdata             raw        511   511   1    closed/syncd  N/A
saplogvg:
LV NAME             TYPE       LPs   PPs   PVs  LV STATE      MOUNT POINT
saplog              raw        255   255   1    closed/syncd  N/A
rootvg:
LV NAME             TYPE       LPs   PPs   PVs  LV STATE      MOUNT POINT
hd5                 boot       1     1     1    closed/syncd  N/A
hd6                 paging     9     9     1    open/syncd    N/A
hd8                 jfs2log    1     1     1    open/syncd    N/A
hd4                 jfs2       1     1     1    open/syncd    /
[MORE...7]

F1=Help             F2=Refresh          F3=Cancel           F6=Command
F8=Image            F9=Shell            F10=Exit            /=Find
n=Find Next
```

Figure 4: logical volumes by volume group (smitty lv)

After both logical volumes are created, it is required that the shared volumes are varied on in an unlock state. This could be done with the command with `varyonvg` using the `-u` option:

`-u` Varies on a volume group, but leaves the disks that make up the volume group in an unlocked state. Use this flag as part of the initial varyon of a dormant volume group.

`-b` Breaks disk reservations on disks locked as a result of a normal varyonvg command. Use this flag on a volume group that is already varied on.

Example:

```
# varyoffvg saplogvg
# varyonvg -u saplogvg
# lsvg -L saplogvg
VOLUME GROUP:       saplogvg                 VG IDENTIFIER:  00cd74fe00004c00000001052baf1b5c
VG STATE:           active                   PP SIZE:        4 megabyte(s)
VG PERMISSION:      read/write               TOTAL PPs:      255 (1020 megabytes)
MAX LVs:            256                      FREE PPs:       0 (0 megabytes)
LVs:                1                        USED PPs:       255 (1020 megabytes)
OPEN LVs:           0                        QUORUM:         2
TOTAL PVs:          1                        VG DESCRIPTORS: 2
STALE PVs:          0                        STALE PPs:      0
ACTIVE PVs:         1                        AUTO ON:        no
MAX PPs per VG:     32512                                    0
MAX PPs per PV:     1016                     MAX PVs:        32
LTG size (Dynamic): 256 kilobyte(s)          AUTO SYNC:      no
HOT SPARE:          no                       BB POLICY:      relocatable
#
```

Now both raw logical volumes for the database are available on the MASTER node.

```
miles[root]/> ls -la /dev/*sap*
crw-rw----    1 root     system       46,  1 Jul 18 16:05 /dev/rsapdata
crw-rw----    1 root     system       45,  1 Jul 18 16:06 /dev/rsaplog
brw-rw----    1 root     system       46,  1 Jul 18 16:05 /dev/sapdata
crw-rw----    1 root     system       46,  0 Jul 18 15:47 /dev/sapdatavg
brw-rw----    1 root     system       45,  1 Jul 18 16:06 /dev/saplog
crw-rw----    1 root     system       45,  0 Jul 18 15:42 /dev/saplogvg
```

```
#
```

The database software will access the raw devices. Hence the owner for the database software needs to have access to the devices. In this example the owner of the database software has the user ID **sdb** e.g. the command will be:

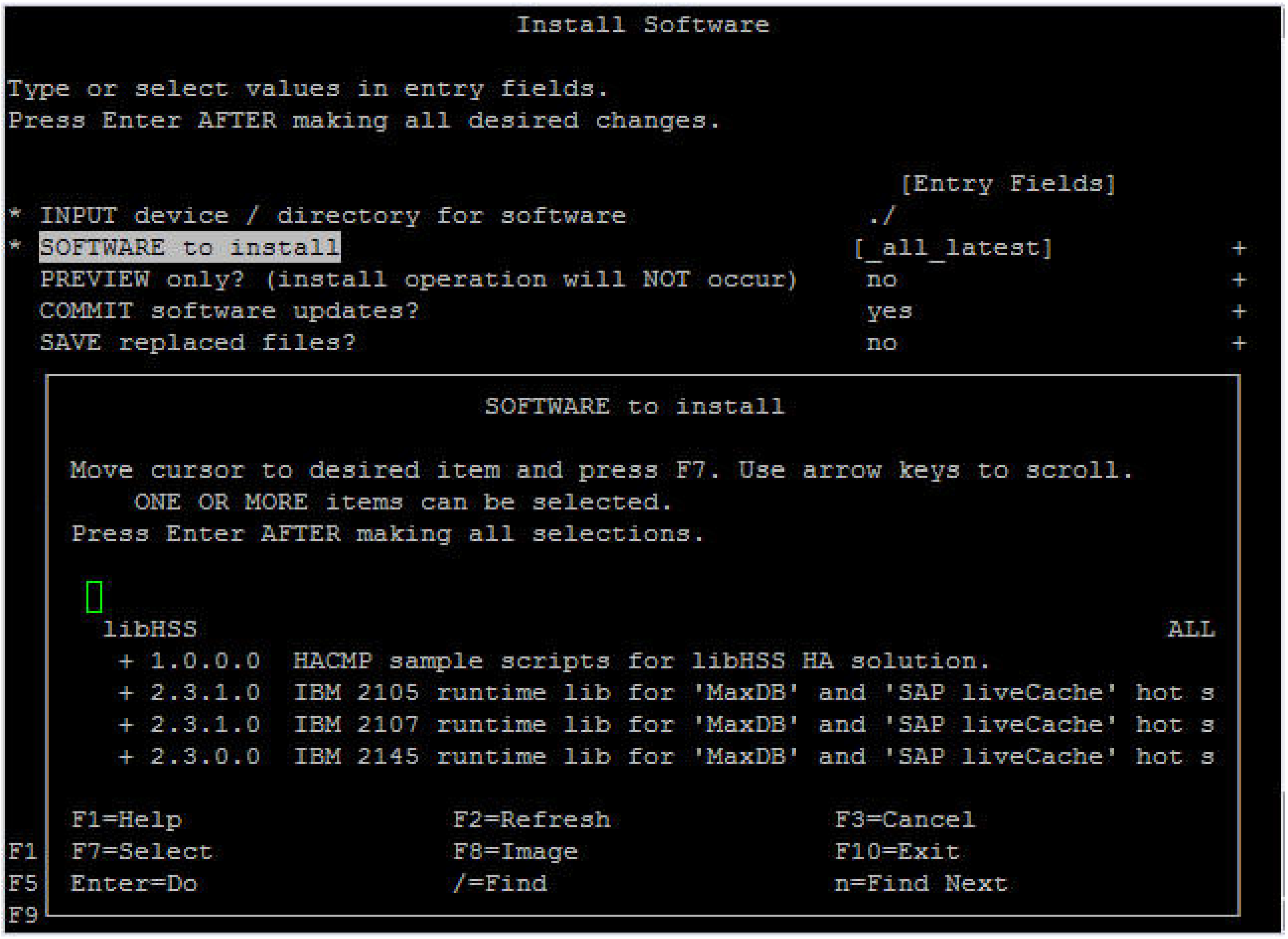
```
# chown sdb /dev/rsap*
# ls -la /dev/*sap*
crw-rw----   1 sdb      system       46,  1 Jul 18 16:05 /dev/rsapdata
crw-rw----   1 sdb      system       45,  1 Jul 18 16:06 /dev/rsaplog
brw-rw----   1 root     system       46,  1 Jul 18 16:05 /dev/sapdata
crw-rw----   1 root     system       46,  0 Jul 18 15:47 /dev/sapdatavg
brw-rw----   1 root     system       45,  1 Jul 18 16:06 /dev/saplog
crw-rw----   1 root     system       45,  0 Jul 18 15:42 /dev/saplogvg
#
```

3.1.2 Install the software library

The library will be installed with either the command line `installp` or using `smitty` software installation. In Figure 5: installing components with smitty the available packages are shown.

- For ESS support choose the package `libHSS.ibm2105.rte ‘IBM 2105 runtime lib for ‘MaxDB’ and ‘SAP liveCache’ hot standby storage’.`
- For DS 8000 support choose the package `libHSS.ibm2107.rte ‘IBM 2107 runtime lib for ‘MaxDB’ and ‘SAP liveCache’ hot standby storage’`
- For SVC support choose the package `libHSS.ibm2145.rte ‘IBM 2145 runtime lib for ‘MaxDB’ and ‘SAP liveCache’ hot standby storage’.`

In addition you could install a couple of sample scripts for the HACMP solution. All files will be copied into a subdirectory of `/usr/opt/ibm/ibmsap` and appropriate subdirectories.



```
                           Install Software

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                           [Entry Fields]
* INPUT device / directory for software                  ./
* SOFTWARE to install                                   [_all_latest]        +
  PREVIEW only? (install operation will NOT occur)       no                  +
  COMMIT software updates?                               yes                 +
  SAVE replaced files?                                   no                  +
  +----------------------------------------------------------------------+
  |                        SOFTWARE to install                           |
  |                                                                      |
  | Move cursor to desired item and press F7. Use arrow keys to scroll.  |
  |     ONE OR MORE items can be selected.                               |
  | Press Enter AFTER making all selections.                             |
  |                                                                      |
  |                                                                      |
  |   libHSS                                                         ALL |
  |    + 1.0.0.0  HACMP sample scripts for libHSS HA solution.           |
  |    + 2.3.1.0  IBM 2105 runtime lib for 'MaxDB' and 'SAP liveCache' hot s |
  |    + 2.3.1.0  IBM 2107 runtime lib for 'MaxDB' and 'SAP liveCache' hot s |
  |    + 2.3.0.0  IBM 2145 runtime lib for 'MaxDB' and 'SAP liveCache' hot s |
  |                                                                      |
  | F1=Help              F2=Refresh             F3=Cancel                |
F1| F7=Select            F8=Image               F10=Exit                 |
F5| Enter=Do             /=Find                 n=Find Next              |
F9+----------------------------------------------------------------------+
```

Figure 5: installing components with smitty

The following files will be copied to the directory `/usr/opt/ibm/ibmsap`

- The HACMP sample scripts for libHSS HA solution

```
/usr/opt/ibm/ibmsap
drwxr-x---   2 root     system          512 Dec 12 18:51 HACMP_samples

/opt/ibm/ibmsap/HACMP_samples:
-rwxr-x---   1 root     system         3177 Jul 14 09:09 hacmpr3.profile
-rwxr-x---   1 root     system         4277 Sep 16 08:00 start_lcappmon
-rwxr-x---   1 root     system         3531 Sep 16 08:00 start_lchot
-rwxr-x---   1 root     system         6724 Sep 16 08:00 start_lcmaster
-rwxr-x---   1 root     system         1003 Dec 12 18:51 start_vserver
-rwxr-x---   1 root     system          939 Sep 16 08:00 stop_lchot
-rwxr-x---   1 root     system          893 Jul 14 09:09 stop_lcmaster
-rwxr-x---   1 root     system          985 Dec 12 18:51 stop_vserver
```

- The IBM 2105 runtime lib

```
/usr/opt/ibm/ibmsap:
-r-xr-xr-x   1 root     system         2805 Mar 16 16:22 RTEHSS_config_2105_sample.txt
-r-xr-x---   1 root     system       200615 Mar 11 14:05 libHSSibm2105.so
drwxrwxrwx   5 root     system          256 Aug 01 18:30 connectors

/usr/opt/ibm/ibmsap/connectors
drwxr-x---   2 root     system          512 Sep 15 15:32 HSS2105

/opt/ibm/ibmsap/connectors/HSS2105:
-rw-r-----   1 root     system         6715 Sep 16 08:57 README_2105_AIX.txt
-rwxrwxrwx   1 sdb      sdba           6538 Aug 01 18:30 README_AIX.txt
```

```
lrwxrwxrwx   1 sdb      sdba             32 Sep 13 14:29 rsExecuteTask.sh ->
/opt/ibm/ESScli/rsExecuteTask.sh
lrwxrwxrwx   1 sdb      sdba             34 Sep 13 14:30 rsQueryComplete.sh ->
/opt/ibm/ESScli/rsQueryComplete.sh
lrwxrwxrwx   1 sdb      sdba             32 Sep 13 14:30 rsRemoveTask.sh ->
/opt/ibm/ESScli/rsExecuteTask.sh #
dr-xr-x---   1 root     system
```

- The IBM 2107 runtime lib

```
/usr/opt/ibm/ibmsap
-rw-rw----   1 root     system         2700 Sep 15 15:31 RTEHSS_config_sample.txt
drwxr-x---   5 root     system          512 Sep 15 15:32 connectors
-r-xr-x---   1 root     system       210179 Sep 15 14:59 libHSSibm2107.so

/usr/opt/ibm/ibmsap/connectors
drwxr-x---   2 root     system          512 Sep 15 15:32 HSS2107

/opt/ibm/ibmsap/connectors/HSS2107:
-rw-r-----   1 root     system         6756 Sep 15 15:55 README_2107_AIX.txt
-rw-r-----   1 root     system         7713 Sep 15 15:15 rsExecuteTask.sh
-rw-r-----   1 root     system         9828 Sep 15 15:15 rsQueryComplete.sh
-rw-r-----   1 root     system         7664 Sep 15 15:15 rsRemoveTask.sh
```

- The IBM 2145 runtime lib

```
/usr/opt/ibm/ibmsap
-rw-rw----   1 root     system         2700 Sep 15 15:31 RTEHSS_config_sample.txt
drwxr-x---   5 root     system          512 Sep 15 15:32 connectors
-r-xr-x---   1 root     system       210179 Sep 15 14:59 libHSSibm2145.so

/usr/opt/ibm/ibmsap/connectors
drwxr-x---   2 root     system          512 Sep 15 15:32 HSS2145

/opt/ibm/ibmsap/connectors/HSS2145:
-rw-r-----   1 root     system         6818 Sep 15 15:55 README_2145_AIX.txt
-rw-r-----   1 root     system        13407 Sep 15 15:15 rsExecuteTask.sh
-rw-r-----   1 root     system        10607 Sep 15 15:15 rsQueryComplete.sh
-rw-r-----   1 root     system         8924 Sep 15 15:15 rsRemoveTask.sh
```

After installing the package, the following steps must be performed.

1. move the storage dependent library file [`libHSSibm2105.so` | `libHSSibm2107.so` | `libHSSibm2145.so`] to the MaxDB/SAP liveCache dependent program path into the shared lib directory [`/<dependent_path>/lib/lib64`]
   If this isn't known, use the dbmcli command: `dbmcli db_enum`. The command shows the path for each installed instance name. Below two examples for:
   open source standard:
   ```
   miles[root]/> /opt/sdb/programs/bin/dbmcli db_enum
   OK
   ESSDB   /opt/sdb/7500   7.5.0.11 fast    offline
   ESSDB   /opt/sdb/7500   7.5.0.11 quick   offline
   ESSDB   /opt/sdb/7500   7.5.0.11 slow    offline
   ```
   SAP standard:
   ```
   enhot1 > dbmcli db_enum
   OK
   HOT     /sapdb/HOT/db   7.5.0.11        fast    offline
   HOT     /sapdb/HOT/db   7.5.0.11        quick   offline
   HOT     /sapdb/HOT/db   7.5.0.11        slow    offline
   ```
2. Change the owner and group rights for the lib to the owner of the database software. If you take the standard settings this will be `sdb` and `sdba`. The file `/etc/opt/sdb`

contain the owner and group of the database software.

(e.g.

```
[Globals]
IndepData=/sapdb/data
IndepPrograms=/sapdb/programs
SdbOwner=sdb
SdbGroup=sdba
```

3. Check the user rights after you moved the files; they should be read and executable for the owner and group.
4. Create a directory with the <SID> name in the directory `/usr/opt/ibm/ibmsap/` (use capitals for the directory name!)
5. Move the example configuration file to this directory e.g.
   `RTEHSS_config_sample.txt`
6. Rename / copy the sample file to `RTEHSS_config.txt`
7. Now edit the `RTEHSS_config.txt` file and make the appropriate changes regarding to your system setup.
8. Since the database software will run with owner and group ID of its owner, this owner must have read and executable rights for the listed directories and all including files.
   The home directory of:
   ESS cli: `/opt/ibm/ESScli` (old standard homedir:`/usr/opt/ibm/ibm2105cli/`)
   DS cli: `/opt/ibm/dscli`
   and the directory `/usr/opt/ibm/ibmsap/`.
   perform one of the following changes to the appropriate directories / files
   a. change the owner and / or group to the owner / group of the database software (e.g. `sdb:sdba`)
   b. add the database software owner to the group `system`
   c. make them read and executable for all

Multiple volume support on storage subsystem

DS: use comma separated string for the variables **MlCDataVdiskID**, **SlCDataVdiskID** and optional **SSlCDataVdiskID**

Example: MlCDataVdiskID 1401,1403 and MlCDataVdiskID 1402,1404 will copy the DS8000 volumes 1401 ⇔ 1402 and 1403 ⇔ 1404 in a single command.

ESS: use consistency groups if the data volumes are spread over more than one ESS volume

SVC: Use comma separated string for the variables **MlCDataVdiskID**, **SlCDataVdiskID** and optional **SSlCDataVdiskID**.

Example: MlCDataVdiskID 5,6,7 and MlCDataVdiskID 8,9,10 will copy the SVC VDiskID's 5 ⇔ 8 and 6 ⇔ 9 and 7 ⇔ 10 as a consistency group.

Below some hints for tailoring the `RTEHSS_config.txt` file:

1. **CSmode** only **FC** could be set
   FC: flash copy only the data volumes, used in case that only one storage system is used
2. **Ibmclidir** directory of the Command Line Interface (CLI)
   DS: default: /opt/ibm/dscli
   ESS: default: /opt/ibm/ESScli

old default: /usr/opt/ibm/ibm2105cli

SVC: default: empty

3. **Ibmsapapodir** install directory of the storage dependent runtime lib [e.g. /usr/opt/ibm/ibmsap]
4. **MlCLogVdiskID**
   DS: ID of volume (four digit hex, e.g. 1400)
   ESS: blank
   SVC: ID of Vdisk (number, e.g. 6)
5. **MlCDataVdiskID**
   DS: ID of volume, for multiple volumes use a comma separated string (e.g. 1401,1403)
   ESS: blank
   SVC: ID of Vdisk, for multiple volumes use a comma separated string (e.g. 5,12,13)
6. **Remaining Nodes variable** the same variables will be used for the first **S**tandby **l**ive**C**ache (SlC) Server and the **S**econd **S**tandby **l**ive**C**ache (SSlC) server.
7. **CSaIP** IP address of
   DS: hmc1
   ESS: copy service server
   SVC: master console
8. **CSaUID**
   DS: user ID (perform copy service task)
   ESS: user ID (admin)
   SVC: ID name for SSH connection to SVC node
9. **CSapwd**
   DS: password of user ID
   ESS: Admin password
   SVC: SSH pass phrase is not supported on SVC nodes, so leave this field blank
10. **CSbIP IP** (user ID and password must be identical for both servers)
    DS: hmc2
    ESS: address of backup copy server
    SVC: backup master console
11. **HSS_NODE_00Y** [Y ε {1, 2, 3}] name for each node. (get the name with 'uname –n' on AIX hosts.
12. **EstDataCST_00X_00Y** [X, Y ε {1, 2, 3}]
    will be used in case of one storage system using flash copy to copy the DATA volumes from HSS_NODE_00X to HSS_NODE_00Y.
    a. DS: sequence number of copy server task. The sequence number is a four digit hexadecimal number (0000 - FFFF). Every task will be build up dynamically.
    b. ESS: the task name to establish the copy server task for DATA volumes. This task must be predefined on the ESS copy service server.
    c. SVC: unique task name to build up the flashCopy relation dynamically.

13. **TermDataCST_00X_00Y** [X, Y ε {1, 2, 3}]
    will be used in case of terminate the flashCopy relation between Volumes of HSS_NODE_00X to HSS_NODE_00Y.
    a. DS: set the identical sequence number as used for establishing the flashCopy relation.
    b. ESS: task name to terminate the flashCopy task
    c. SVC: set the identical name as it was used to establish the flashCopy

3.1.3 Create the database instance

Use the common SAP tools SAPINST or SDBINST for the MaxDB Software installation. SAPINST or the MaxDB Database Manager (e.g. DBMGUI) creates a MaxDB or liveCache instance.

Use the previous defined raw devices for log and data volumes.

Please find the MaxDB Hot Standby Documentation at
http://help.sap.com/saphelp_nw04s/helpdata/en/70/57d43fdd561165e10000000a114b1d/frameset.htm

3.2 HSS_NODE_002 => STANDBY

3.2.1 Set up the volumes

Since all copy functions of the Storage Systems are making an exact physical copy (e.g. sector by sector) of the complete disk, the logical volume information is also copied one-to-one. This means, all standby nodes will have identical logical raw volumes (e.g. in case of one storage system a shared LOG and copied DATA), the Physical Volume ID (PVID) for each volume will be identical on all systems! This means we could not use the `receratevg` command to define the volumes on the other nodes. Instead of this we need to use `importvg`.

To import the volumes, we need to have the logical information about the Volume Group (VG) and the Logical Volume (LV) available on the standby nodes. For all volumes which will be copied later on automatically (using the flashCopy function), we need to do the copy once before we could set up the volumes on the standby. This step is different for each storage system and is therefore described in the next chapter.

Lock how the system is configured so far. Use the `lsvpcfg` command to see the storage system disks.

Master:

```
# pcmpath query device

DEV#:   4  DEVICE NAME: hdisk4  TYPE: 2107900  ALGORITHM:  Load Balance
SERIAL: 75730411400
==========================================================================
Path#      Adapter/Path Name          State     Mode     Select     Errors
    0           fscsi0/path0           OPEN   NORMAL       4466          0
    1           fscsi0/path1           OPEN   NORMAL       4588          0
    2           fscsi1/path2           OPEN   NORMAL       4469          0
    3           fscsi1/path3           OPEN   NORMAL       4508          0

DEV#:   5  DEVICE NAME: hdisk5  TYPE: 2107900  ALGORITHM:  Load Balance
SERIAL: 75730411401
==========================================================================
Path#      Adapter/Path Name          State     Mode     Select     Errors
```

```
    0           fscsi0/path0           OPEN   NORMAL       183          0
    1           fscsi0/path1           OPEN   NORMAL       191          0
    2           fscsi1/path2           OPEN   NORMAL       195          0
    3           fscsi1/path3           OPEN   NORMAL       167          0
# lspv
hdisk0          005e4c1c05fdaed0                    None
hdisk1          005e4c1c4f4eca5c                    None
hdisk2          00cd74fe841990fc                    rootvg          active
hdisk3          00cd74fece77fd41                    rootvg          active
hdisk4          00cd74fe2baf0789                    saplogvg        active
hdisk5          00cd74fe2bb3ca40                    sapdatavg       active
#
```

Standby:

First copy the storage volume which will hold the SAP liveCache / MySQL MaxDB data volumes of the master node to the volume which will be assigned to the standby node. To perform the copy, refer to the next chapter, depending by the used storage system. As soon as the volume(s) are copied, run the configuration manager `cfgmgr` on the standby node. The `pcmpath query device` command will show the DS volumes assigned to the host connection for the standby node and the `lspv` command shows that the volumes already having a pvid.

```
# cfgmgr
# pcmpath query device

DEV#:   4  DEVICE NAME: hdisk4  TYPE: 2107900  ALGORITHM:  Load Balance
SERIAL: 75730411400
==========================================================================
Path#      Adapter/Path Name          State     Mode     Select     Errors
    0          fscsi0/path0          CLOSE   NORMAL          0          0
    1          fscsi0/path1          CLOSE   NORMAL          0          0
    2          fscsi1/path2          CLOSE   NORMAL          0          0
    3          fscsi1/path3          CLOSE   NORMAL          0          0

DEV#:   5  DEVICE NAME: hdisk5  TYPE: 2107900  ALGORITHM:  Load Balance
SERIAL: 75730411402
==========================================================================
Path#      Adapter/Path Name          State     Mode     Select     Errors
    0          fscsi0/path0          CLOSE   NORMAL          0          0
    1          fscsi0/path1          CLOSE   NORMAL          0          0
    2          fscsi1/path2          CLOSE   NORMAL          0          0
    3          fscsi1/path3          CLOSE   NORMAL          0          0
# lspv
# lspv
hdisk0          none                None
hdisk1          none                None
hdisk2          none                None
hdisk3          00cd74eecab750ca                    rootvg          active
hdisk4          00cd74fe2baf0789                    None
hdisk5          00cd74fe2bb3ca40                    None
#
```

During the next step the volume groups for the log and data volumes need to be configured.

3.2.1.1 Copy volumes on the ESS

In case of the ESS run the copy server task you have already defined to copy the data volumes from the MASTER instance to the STANDBY instance. If you need to create the tasks by yourself, see chapter **4.1.2 Create Copy Server tasks** on page **30** to find more information and references.

3.2.1.2 Copy volumes on the SVC

In case of the SVC you could use the SVCCLI (see chapter 4.3 SAN Volume Controller on page **30** for more details).

Login with the MAX DB manager ID and send the SVCCLI commands using SSH. Below this is shown for the sample configuration.

Create flashCopyMapping:

```
$ ssh -l dizzy admin@9.155.50.64 svctask mkfcmap -source 6 -target 3 -name SAP_1_2_EST
FlashCopy Mapping, id [0], successfully created
```

Prepare and start FlashCopyMapping:

```
$ ssh -l dizzy admin@9.155.50.64 svctask startfcmap -prep SAP_1_2_EST
```

Show the progress (`lsfcmapprogress`)of the FlashCopyMapping:

```
$ ssh -l dizzy admin@9.155.50.64 svcinfo lsfcmapprogress SAP_1_2_EST
id              progress
0               5
$
```

PLEASE NOTE: The SVC supports only one mapping per volume.

It is recommended to remove the FlashCopyMapping after its completion!

Execute the remove FlashCopyMapping (`rmfcmap`) command, e.g. type:

```
$ ssh -l dizzy admin@9.155.50.64 svctask rmfcmap SAP_1_2_EST
```

As soon as the mapping was started you could go forward and import the volume groups.

3.2.1.3 Copy volumes on the DS8000

In case of the DS 8000 you could use the dscli (see chapter **4.2 DS 8000** on page **30** for more details).

Login with the MAX DB manager ID (e.g. dbm) and call the dscli. Below this is shown for the sample configuration.

Connect with the dscli to the DS 8000:

```
# cd /opt/ibm/dscli
# ./dscli -user USER_ID -passwd PASSWORD -hmc1 IP-ADRESS
Date/Time: July 20, 2005 2:06:31 PM CDT IBM DSCLI Version: 5.0.3.150  DS:
IBM.2107-7573041

dscli>
```

and perform the single steps as mentioned below or issue the commands in single shoot mode.

Create the flashCopy:

```
# /opt/ibm/dscli/dscli -user USER_ID -passwd PASSWORD -hmc1 IP-ADRESS mkflash -dev IBM.2107-
7573041 -seqnum 1020 1401:1402
```

Show the out of sync tracks (`lsflash -l`) of the flashCopy:

```
# /opt/ibm/dscli/dscli -user USER_ID -passwd PASSWORD -hmc1 IP-ADRESS lsflash -dev IBM.2107-
7573041 -l 1401:1402
```

If the flashCopy was created with the option `-persist`, it is necessary to remove the flashCopy after its completion!

Execute the remove flashCopy (`rmflash`) command, e.g. type:

```
# /opt/ibm/dscli/dscli -user USER_ID -passwd PASSWORD -hmc1 IP-ADRESS rmflash -dev IBM.2107-
7573041 -seqnum 1020 1401:1402
```

As soon as the mkflash command returns successful, the volume groups on the standby node could be imported.

3.2.1.4 Import volume groups

Remember, the copy process makes a one-to-one copy of the physical layer. If the command `cfgmgr` was issued on the standby node before the flasCopy was started, the STANDBY node will not be able to import the vg since the physical volume ID was changed.
The error will lock like:

```
# importvg -n -R -y sapdatavg hdisk5
0516-304 getlvodm: Unable to find device id 0034921afd56d9be in the Device
        Configuration Database.
0516-022 : Illegal parameter or structure value.
0516-780 importvg: Unable to import volume group from hdisk4.
#
```

To fix this, delete the appropriate hdisks and run the configmanager (`cfgmgr`) command.
Example:

```
# rmdev -l hdisk5 -dR
hdisk5 deleted
# cfgmgr
# lspv
hdisk0          none                None
hdisk1          none                None
hdisk2          none                None
hdisk3          00cd74eecab750ca                    rootvg          active
hdisk4          00cd74fe2baf0789                    None
hdisk5          00cd74fe2bb3ca40                    None
# importvg -n -R -y sapdatavg hdisk5
sapdatavg
#
```

Now the system has the correct information and will be able to import the VG

To make all logical information available on the standby node, import the information to the operating system. On AIX use the command `importvg` with the option

`-n` causes the volume not to be varied on at the completion of the volume group import into the system.

`-R` Restores the ownership, group ID, and permissions of the logical volume special device files. These values will be restored only if they were set using U, G and P flags of mklv and chlv commands. This flag is applicable only for big vg format volume groups only

Example:

```
# importvg -n -R -y saplogvg hdisk4
saplogvg
#
```

now check the configuration with `lspv` (list physical volume information)

```
# lspv
hdisk0          none                None
hdisk1          none                None
hdisk2          none                None
hdisk3          00cd74eecab750ca                    rootvg          active
hdisk4          00cd74fe2baf0789                    saplogvg        active
hdisk5          00cd74fe2bb3ca40                    sapdatavg       active
#
```

The volume groups were imported and are available.

3.2.1.5 Varyon the volume groups

Since the volume groups are known on the standby node we could activate them. At AIX use the `varyon` command as already described in chapter **3.1.1 Set up the volumes** on page **5**.

3.2.1.6 Change owner of raw device

Last thing to do is to change the ownership of the logical raw devices from the LOG and DATA volumes. In our example (remember the owner of the database software:

```
# chown sdb /dev/rsap*
# ls -la /dev/*sap*
crw-rw----   1 sdb      sdba         46,  1 Aug 03 12:47 /dev/rsapdata
crw-rw----   1 sdb      sdba         47,  1 Aug 02 14:25 /dev/rsaplog
brw-rw----   1 root     system       46,  1 Jul 26 14:29 /dev/sapdata
crw-r-----   1 root     system       46,  0 Jul 26 14:29 /dev/sapdatavg
brw-rw----   1 root     system       47,  1 Jul 26 14:30 /dev/saplog
crw-r-----   1 root     system       47,  0 Jul 26 14:30 /dev/saplogvg
#
```

3.2.2 Install the software library on the STANDBY

You need to perform the identical tasks as described in chapter **3.1.2 Install the software library** on page **9**. It is recommended to copy the configuration file from the first node to the second one.

3.2.3 Create the database instance

Use some dbmcli commands or the Database Manager GUI to configure the hot standby environment and create the standby instance.
Install the MaxDB- or liveCache - software with SDBINST. Find the documentation of SDBINST under http://help.sap.com/ -> SAP Netweaver -> SAP Web Applicvation Server -> SAP Netweaver Components -> MaxDB -> Installation -> Database Software Installation Guide
(http://help.sap.com/saphelp_nw04s/helpdata/en/4c/da3b3c51bd4b3ae10000000a114084/frameset.htm)
Make sure the X Server is running by starting

```
$ /sapdb/programs/bin/x_server start
```

A description for the most recent commands for the hot standby system is explained. The complete reference is available at
http://www.mysql.com/products/maxdb/pdf/MaxDB.HotStandbySystem.pdf

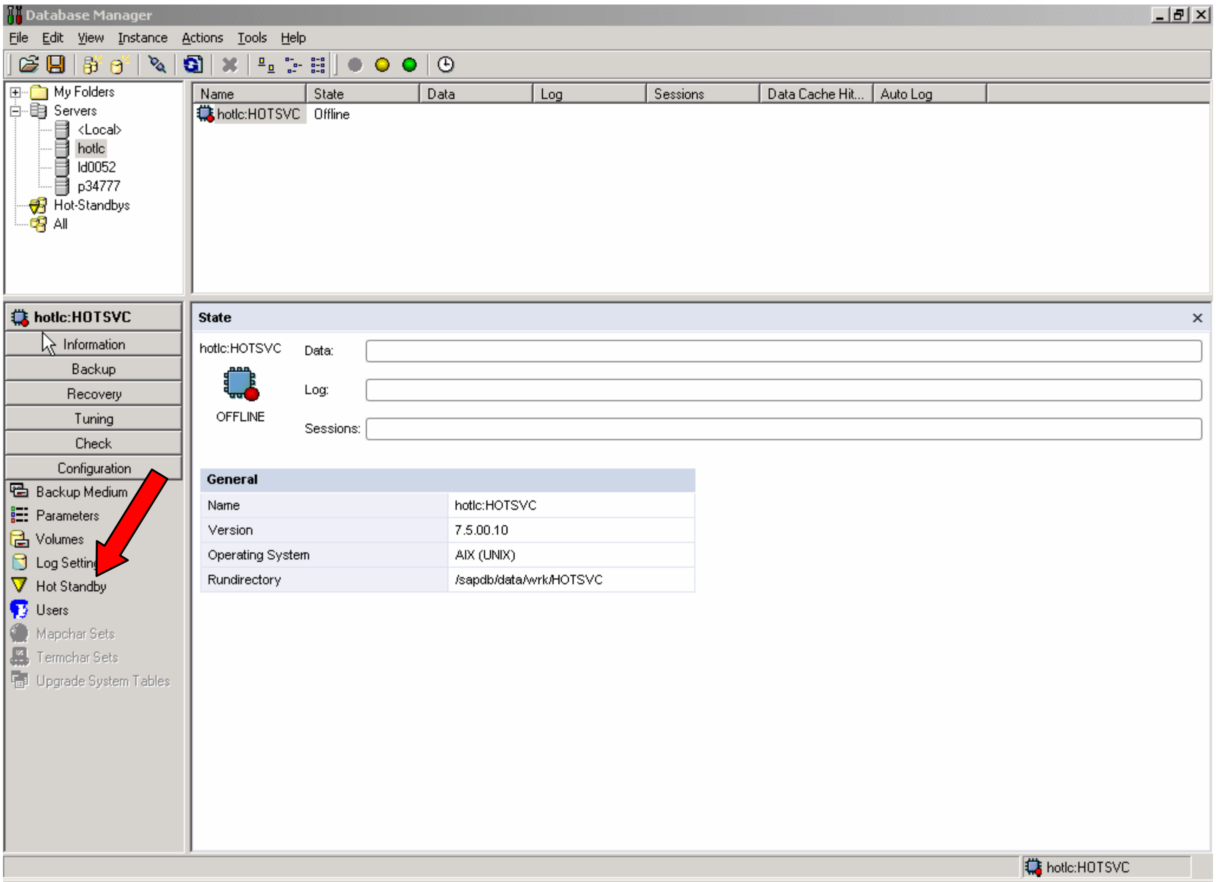
3.2.3.1 Set the master HSS configuration

Execute the HSS integration commands on the master database. First make the master database known as a hot standby instance. Either use dbmcli commands or use the Configuration Wizard of the DBMGUI.

```
$dbmcli –n p520_TIC3 –d <db_name> –u <dbm-user>,<password>
$ dbmcli on p520_TIC3: <dbname> > db_offline
$ dbmcli on p520_TIC3: <dbname> > param_directput ALLOW_MULTIPLE_SERVERTASKS_UKTS YES
$ dbmcli on p520_TIC3: <dbname> > param_checkall
$ dbmcli on p520_TIC3: <dbname> > hss_enable lib=libHSSibm2107 node=<official node>
$ dbmcli on p520_TIC3: <dbname> > db_online
```

The parameter ALLOW_MULTIPLE_SERVERTASKS_UKTS=YES allows rapid log redo in the standby instance. <official node> is the node name the application will use. It's identical to the service name in HACMP.

For DBMGUI register the database instance using the <official node>. Call the Configuration Wizard in the main screen with a double click on **Configuration** >> **Hot Standby**



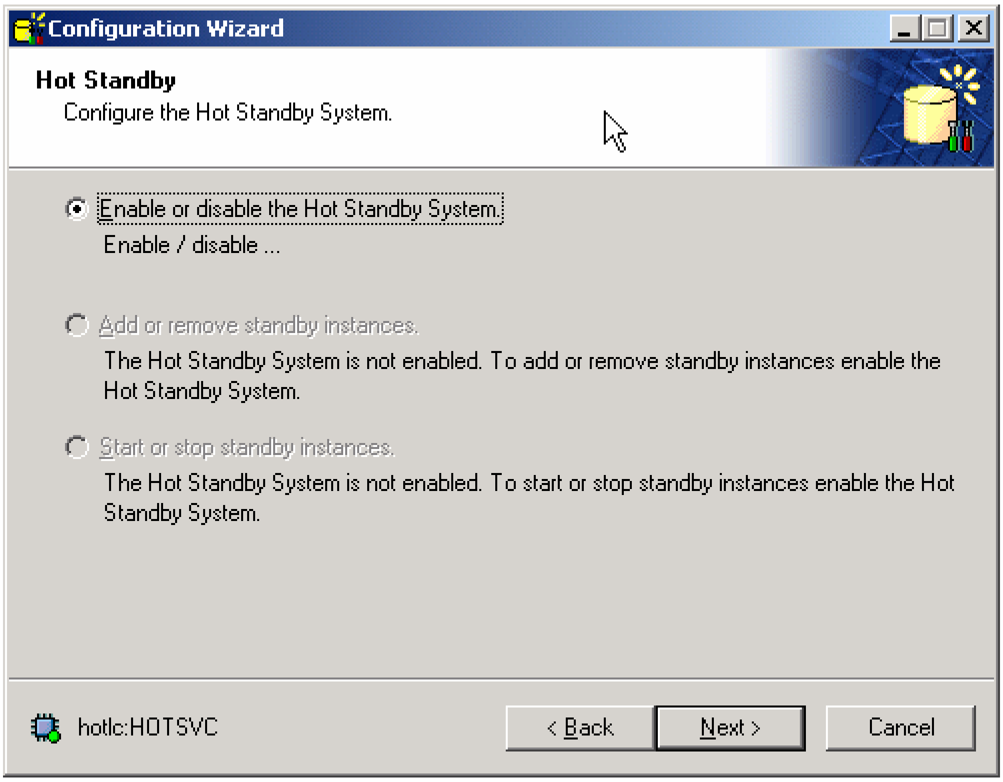
Name	State	Data	Log	Sessions	Data Cache Hit...	Auto Log
hotlc:HOTSVC	Offline					

State: hotlc:HOTSVC — OFFLINE
Data:
Log:
Sessions:

General	
Name	hotlc:HOTSVC
Version	7.5.00.10
Operating System	AIX (UNIX)
Rundirectory	/sapdb/data/wrk/HOTSVC

Figure 6: DBMGUI / Call the Configuration Wizard

Inside the Configuration Wizard choose „Enable or disable the Hot Standby System".



Configuration Wizard

Hot Standby

Configure the Hot Standby System.

● Enable or disable the Hot Standby System.

Enable / disable ...

○ Add or remove standby instances.

The Hot Standby System is not enabled. To add or remove standby instances enable the Hot Standby System.

○ Start or stop standby instances.

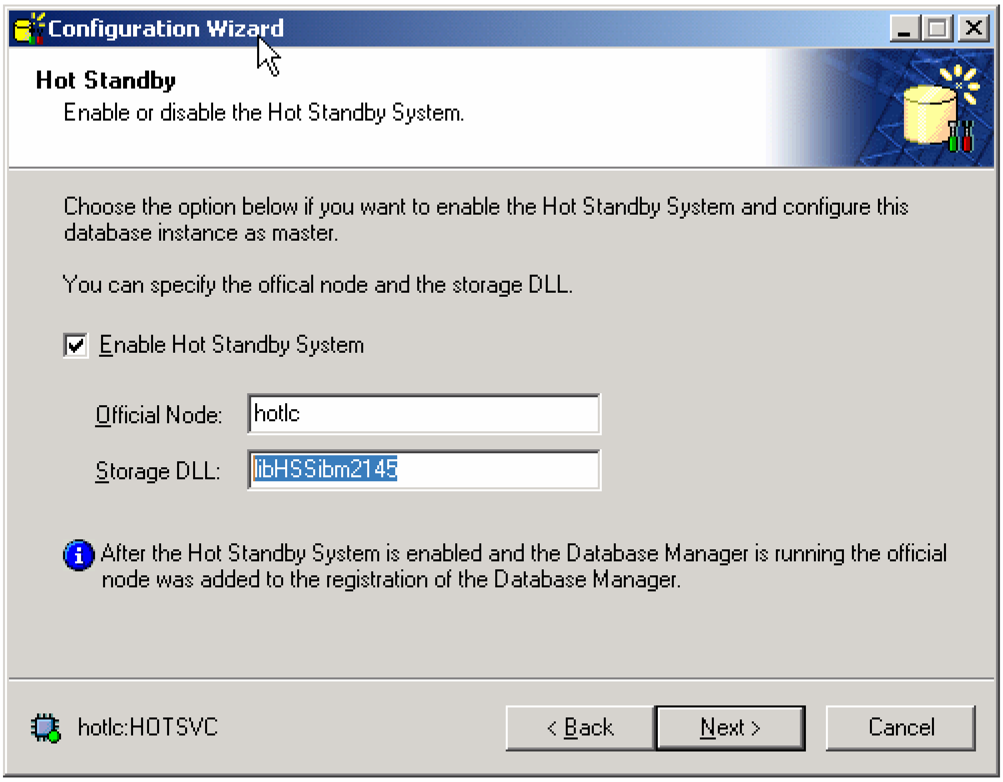
The Hot Standby System is not enabled. To start or stop standby instances enable the Hot Standby System.

hotlc:HOTSVC

Figure 7: DBMGUI / Enable or disable the Hot Standby System

Use **Next,** choose "Enable Hot Standby System" and specify the Official Node with the name of the of the IBM runtime lib for MaxDB and SAP liveCache which is for the

ESS:	libHSSibm2105;
DS8000:	libHSSibm2107;
SVC:	libHSSibm2145.



Hot Standby

Enable or disable the Hot Standby System.

Choose the option below if you want to enable the Hot Standby System and configure this database instance as master.

You can specify the offical node and the storage DLL.

☑ Enable Hot Standby System

Official Node: hotlc

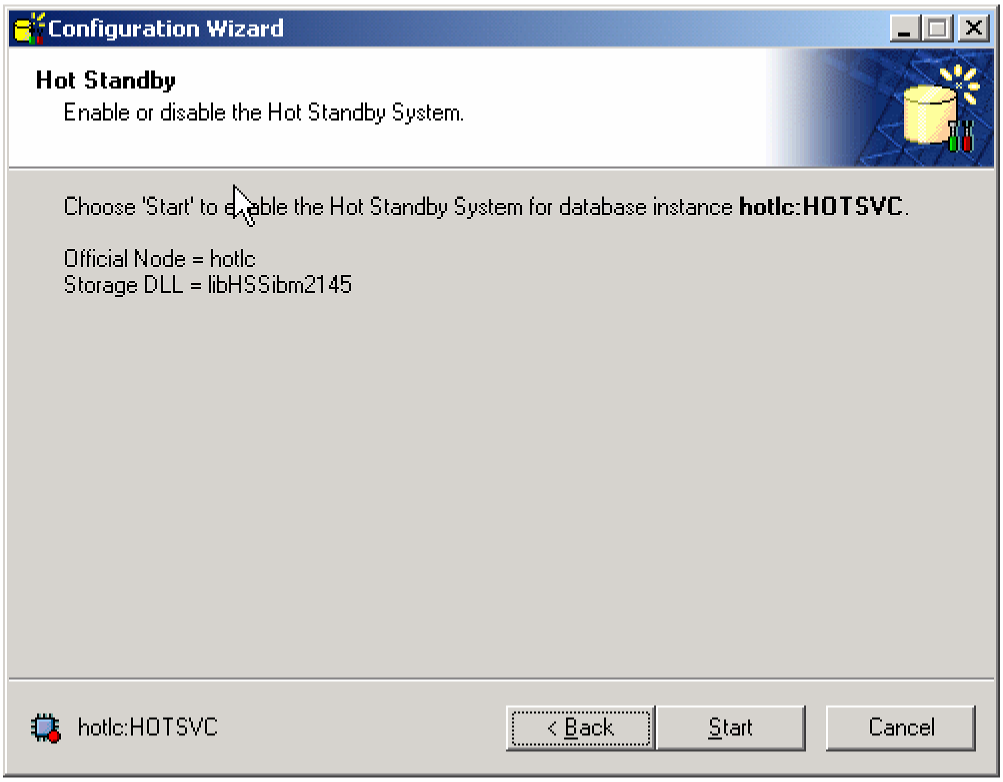
Storage DLL: libHSSibm2145

After the Hot Standby System is enabled and the Database Manager is running the official node was added to the registration of the Database Manager.

hotlc:HOTSVC

Figure 8: DBMGUI / Specify the Official Node with the name of the of the IBM runtime lib

Choose **Next** again and confirm the settings with a click on **Start**.



Configuration Wizard

Hot Standby

Enable or disable the Hot Standby System.

Choose 'Start' to enable the Hot Standby System for database instance **hotlc:HOTSVC**.

Official Node = hotlc
Storage DLL = libHSSibm2145

hotlc:HOTSVC

Figure 9: DBMGUI / Confirm the settings

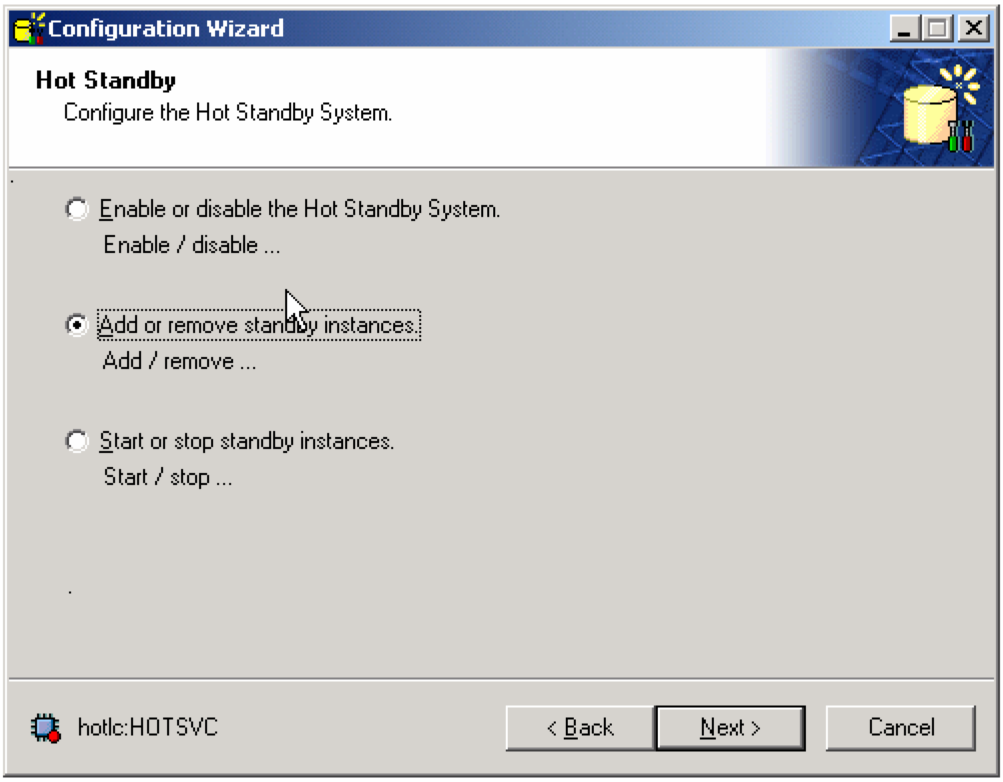
3.2.3.2 Add the standby instance

The dbmcli-command hss_addstandby makes a new instance known to the hot standby environment.

```
$dbmcli –n p520_TIC3 –d <db_name> –u <dbm-user>,<password>
$ dbmcli on p520_TIC3: <dbname> > hss_addstandby p520_TIC4 login=sqdsid,passwd
```

The command hss_addstandby needs the permission to create an instance on the second node. Provide the user and password of the operating system user of the MaxDB or SAP liveCache software.

The same functionality is available in the Configuration Wizard of the DBMGUI. Choose “Add or Remove standby instance” and **Next**.



Configuration Wizard

Hot Standby

Configure the Hot Standby System.

○ Enable or disable the Hot Standby System.
Enable / disable ...

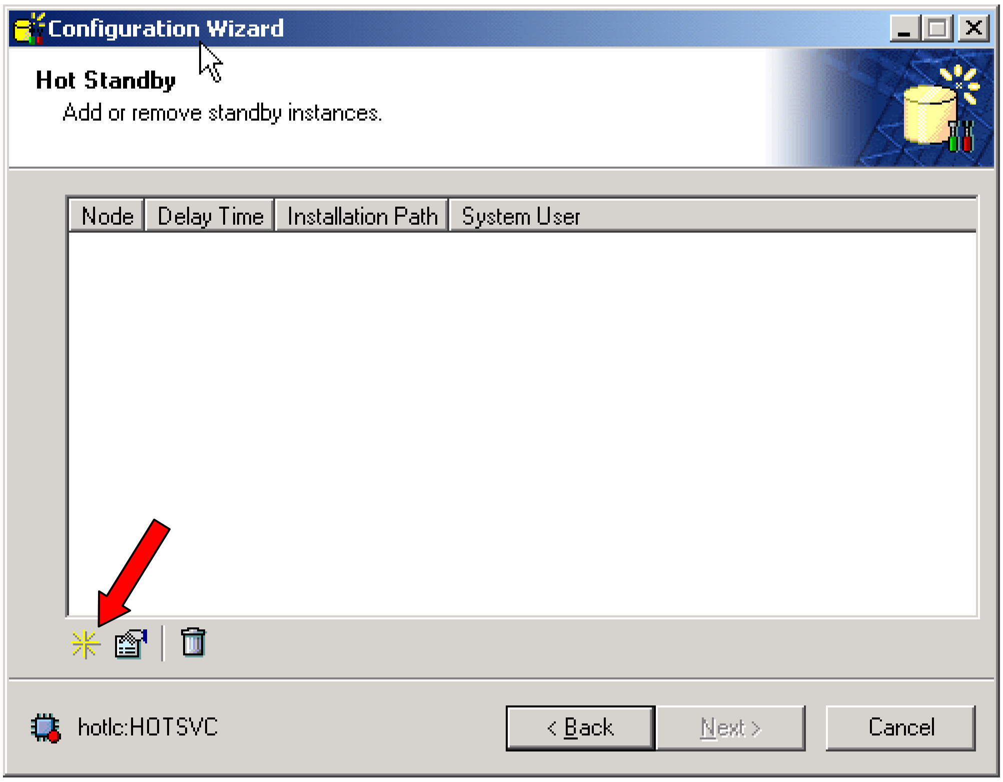
● Add or remove standby instances.
Add / remove ...

○ Start or stop standby instances.
Start / stop ...

hotlc:HOTSVC

Figure 10: DBMGUI / Add or remove standby instance

The displayed table will be empty because no standby instances have been added to the system, yet. Click the asterisk to add a new standby instance.



Configuration Wizard

Hot Standby

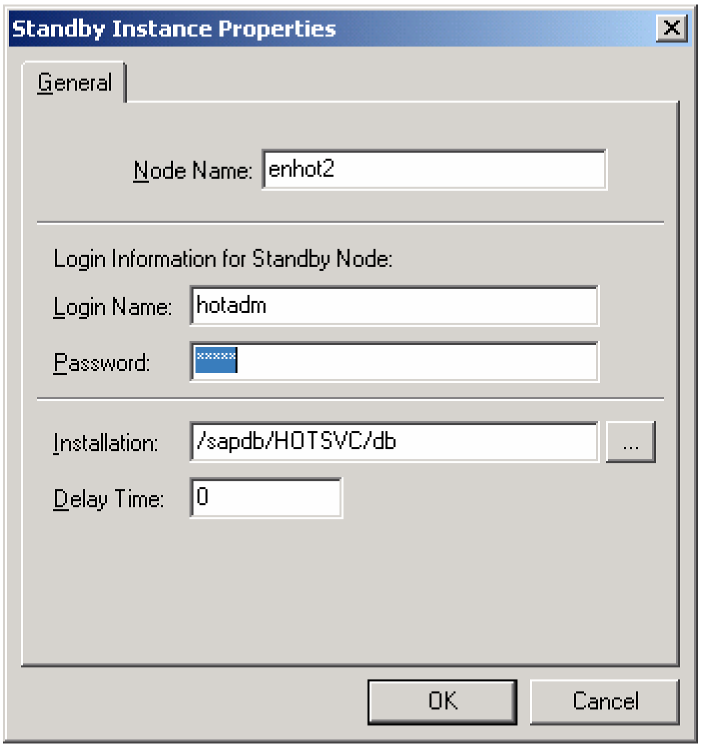
Add or remove standby instances.

Node	Delay Time	Installation Path	System User

hotlc:HOTSVC

Figure 11: DBMGUI / Add a new standby instance

Specify the node name of the server where to install the standby instance, the user and password of the operating system user and the dependent program path on the standby server. Confirm the settings with **OK.**



Standby Instance Properties

Node Name: enhot2

Login Information for Standby Node:

Login Name: hotadm

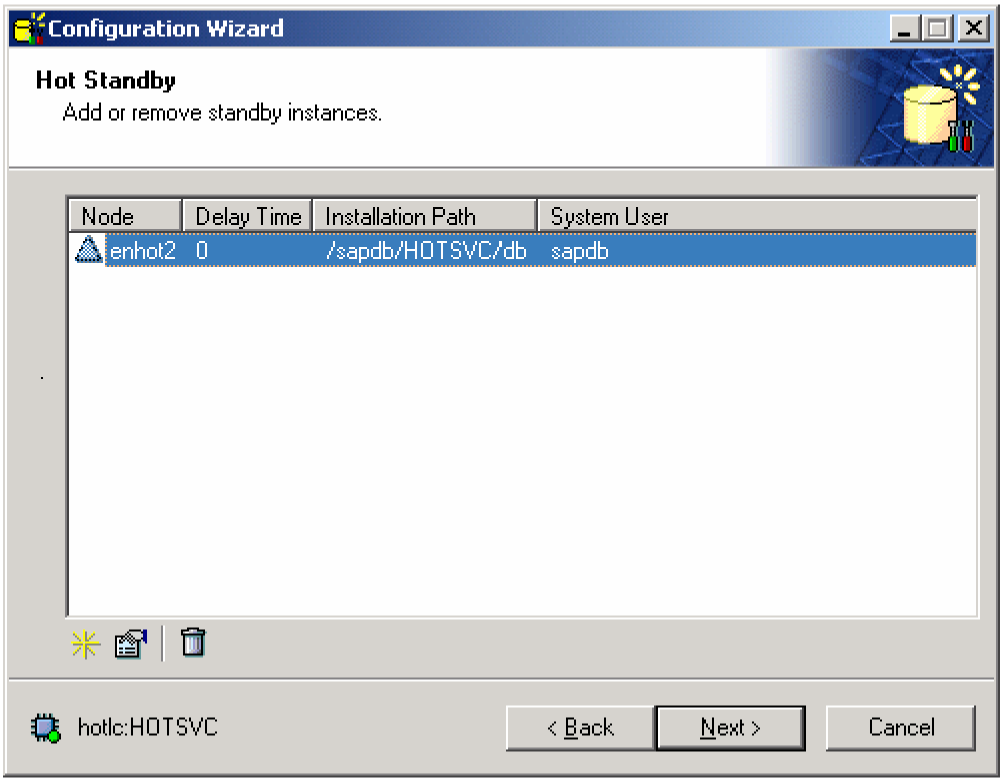
Password: *****

Installation: /sapdb/HOTSVC/db

Delay Time: 0

Figure 12: DBMGUI / Specify standby instance

The table now will show the added instance.



Configuration Wizard

Hot Standby

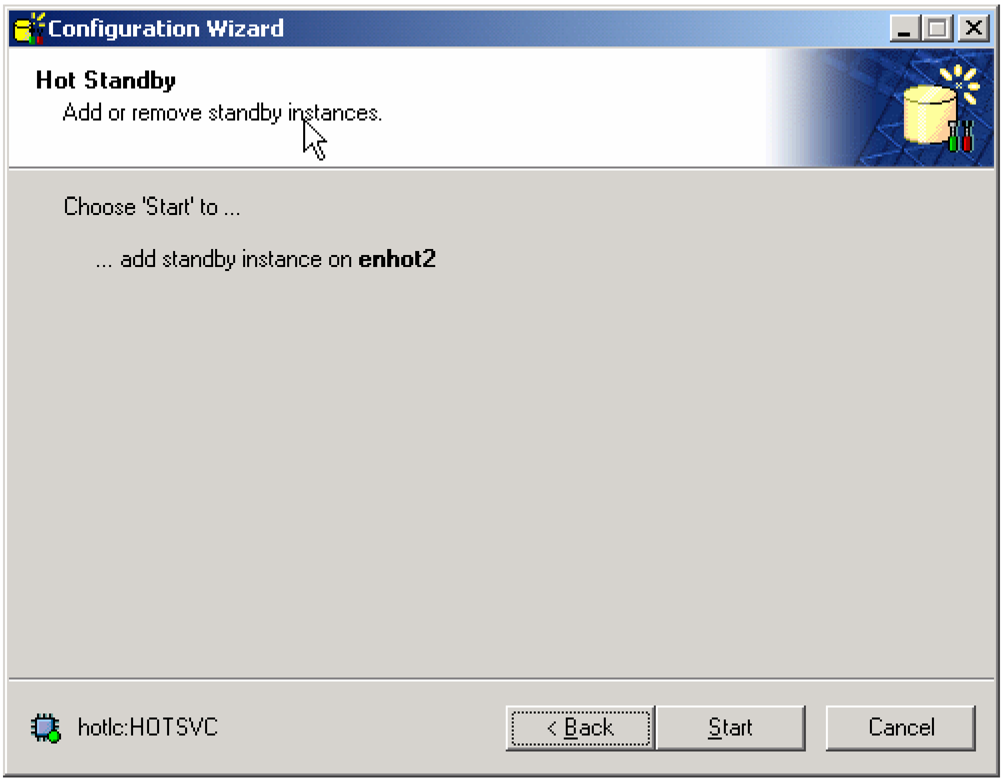
Add or remove standby instances.

Node	Delay Time	Installation Path	System User
enhot2	0	/sapdb/HOTSVC/db	sapdb

hotlc:HOTSVC

Figure 13: DBMGUI / Verify added instance

Click **Next** and confirm the settings with **Start** on the next screen.



Configuration Wizard

Hot Standby

Add or remove standby instances.

Choose 'Start' to ...

... add standby instance on **enhot2**

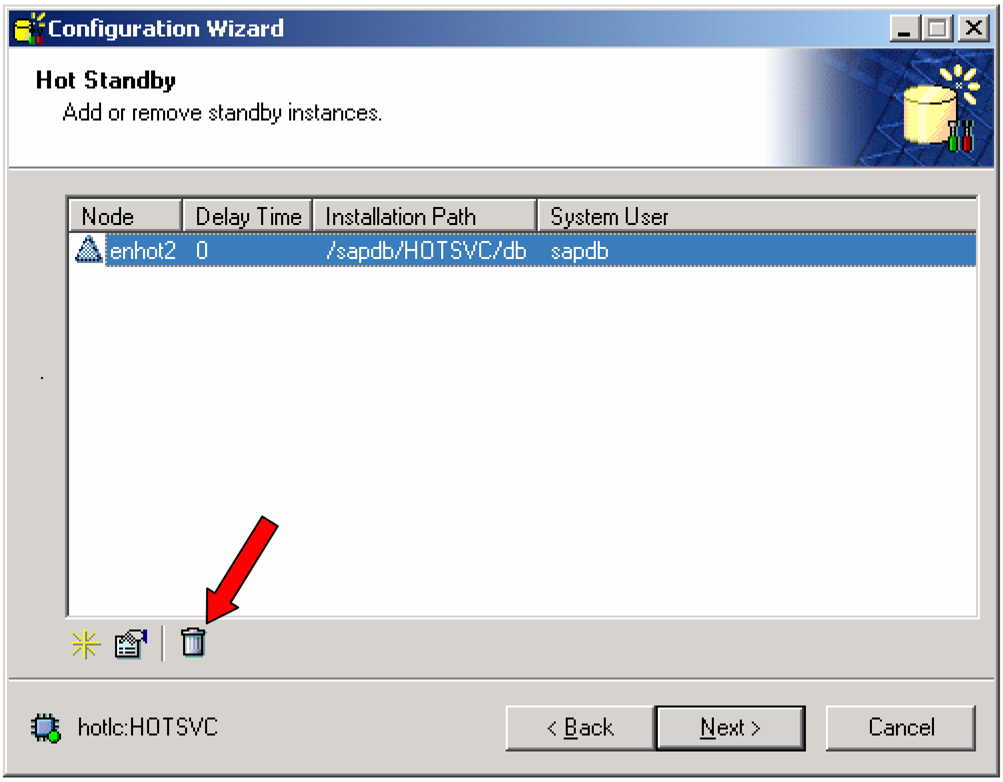
Figure 14: DBMGUI / Activate instance

3.2.3.3 Remove a standby instance

The dbmcli command hss_removestandby removes an instance from the hot standby system. It clears all information about the standby instance from the current master.

```
$dbmcli –n p520_TIC3 –d <db_name> –u <dbm-user>,<password>
$ dbmcli on p520_TIC3: <dbname> > hss_removestandby p520_TIC4
```

Use the selection “Add or Remove standby instance” and **Next** in the Configuration Wizard and click the trash can if you like to remove a standby instance via the DBMGUI.



Hot Standby

Add or remove standby instances.

Node	Delay Time	Installation Path	System User
enhot2	0	/sapdb/HOTSVC/db	sapdb

hotlc:HOTSVC

Figure 15: DBMGUI / Remove standby instance

3.2.3.4 Activate the standby instance

From now on use only the <official node> as server name for the communication to both, master and standby instance. One single dbmcli-command activates the standby instance. Make sure the standby instance is offline.

```
$dbmcli -n <official node> -d <db_name> -u <dbm-user>,<password>
$ dbmcli on <official node>: <dbname> > hss_execute dizzy db_offline
$ dbmcli on <official node>: <dbname> > db_standby dizzy
```

With db_standby the standby instance checks, if the current data volumes fit to the log. If yes, it starts with the redo of the log and returns OK.

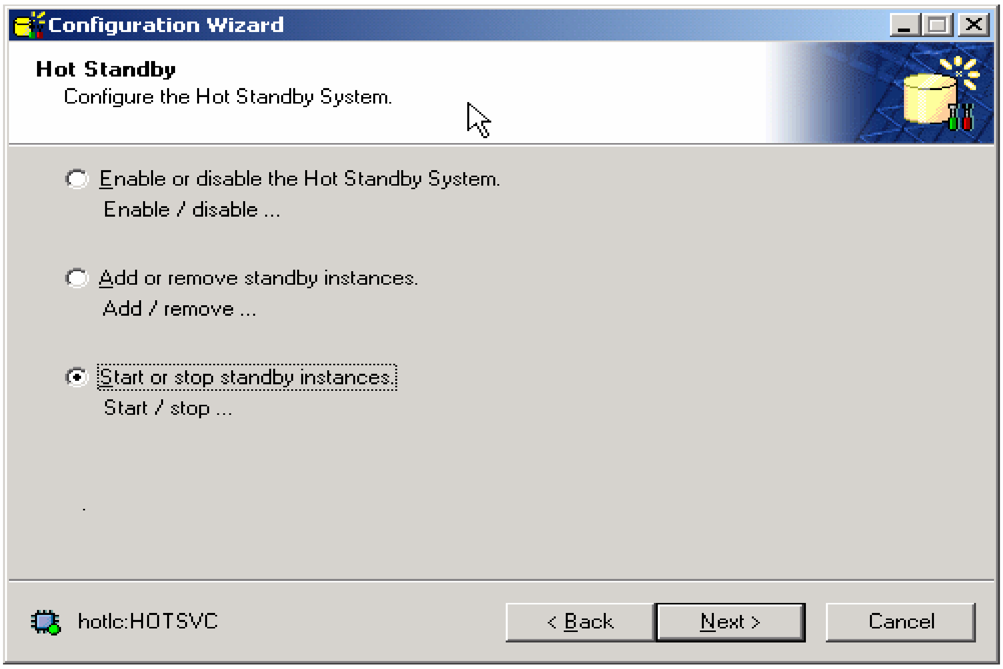
If not, it starts an Init Standby. The Init Standby starts the ESS copy server tasks which copy the data volumes of the master to the standby. This step runs asynchronously.

The db_standby sets the standby instance into mode Standby when the Init Standby returns OK.

A Takeover doesn't need to wait until the copy of the data volumes has finished. Even several take over from master to standby and back are possible.

A second Init Standby would have to wait until the first copy of the data volumes finished, if the initialization will lead to a reverse copy process. If the Init Standby starts a copy process in the same direction as the current process, the current copy process will be stopped and a new flashCopy will be taken from the source volume.

The Configuration Wizard provides the same functionality as the dbmcli command db_standby. Choose “Start or stop standby instance” and confirm with **Next**.


Configuration Wizard

Hot Standby

Configure the Hot Standby System.

○ Enable or disable the Hot Standby System.
Enable / disable ...

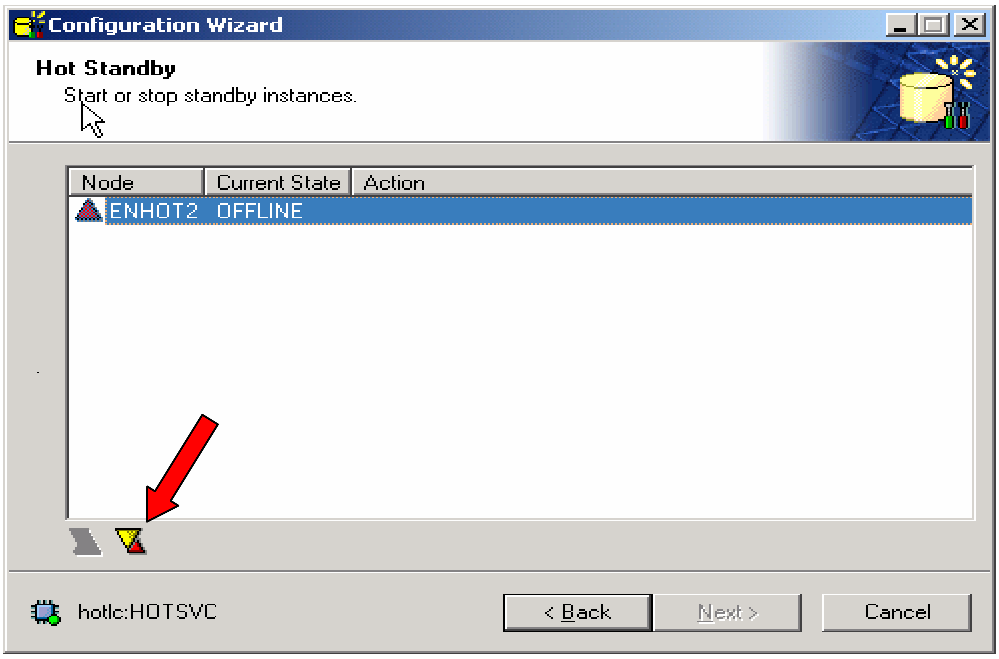
○ Add or remove standby instances.
Add / remove ...

● Start or stop standby instances.
Start / stop ...

hotlc:HOTSVC

Figure 16: DBMGUI / Start or stop standby instance

You will first find a table with all known standby instances and their state. Mark the instance and use the highlighted button to start or stop it.


Configuration Wizard

Hot Standby

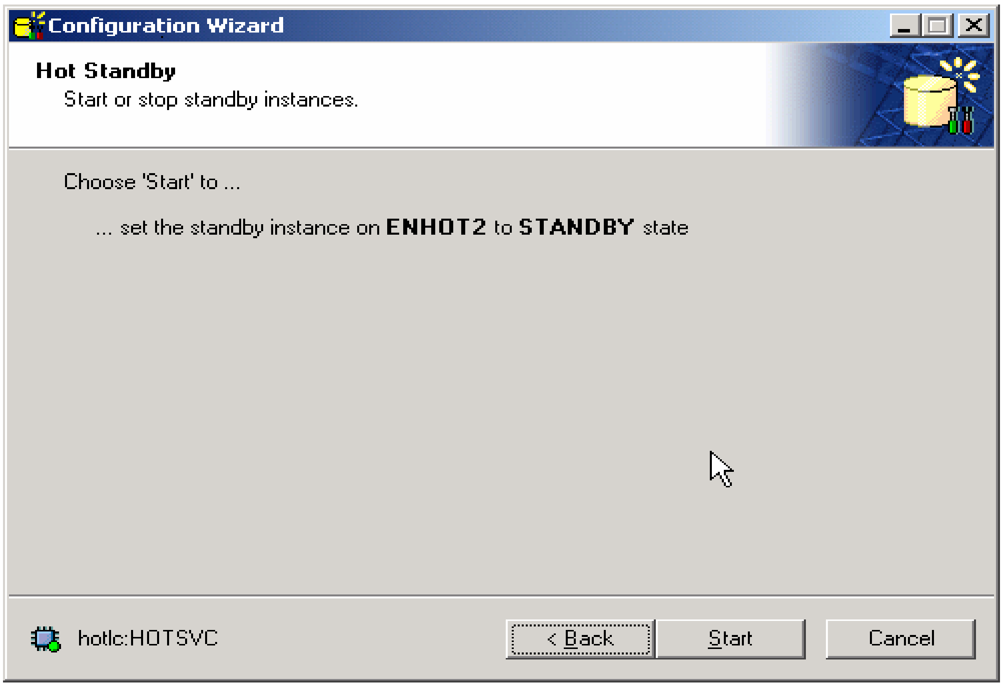
Start or stop standby instances.

Node	Current State	Action
ENHOT2	OFFLINE	

hotlc:HOTSVC

Figure 17: DBMGUI / Choose instance to start/stop it

Confirm the action with a click on **Start**.



Hot Standby

Start or stop standby instances.

Choose 'Start' to ...

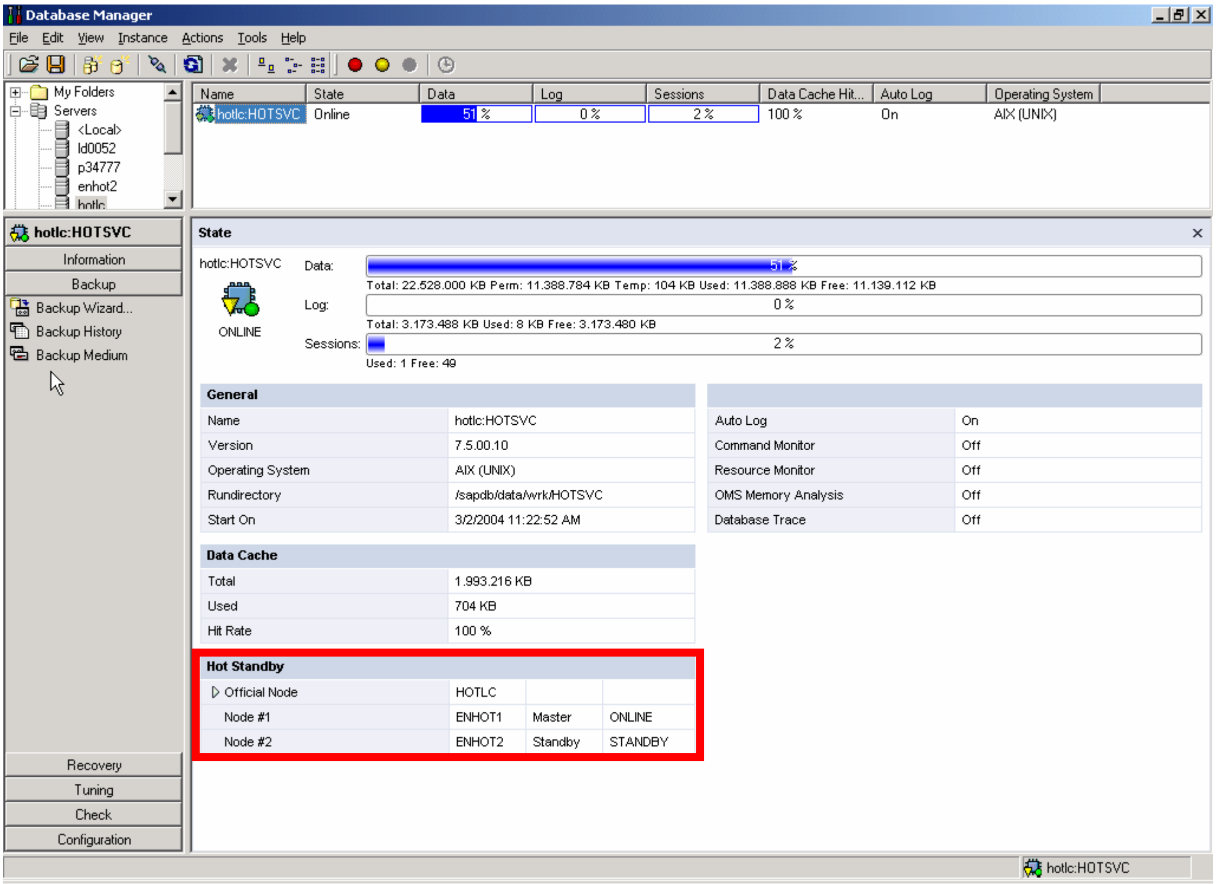
... set the standby instance on **ENHOT2** to **STANDBY** state

hotlc:HOTSVC

Figure 18: DBMGUI / Confirm action

Don’t forget to start the standby instances after each restart of the master. The start of the master doesn’t automatically synchronize the standby instances.

The main screen of the DBMGUI shows information about the master and standby instances.



Name	State	Data	Log	Sessions	Data Cache Hit...	Auto Log	Operating System
hotlc:HOTSVC	Online	51 %	0 %	2 %	100 %	On	AIX (UNIX)

State

hotlc:HOTSVC

ONLINE

Data: 51 %

Total: 22.528.000 KB Perm: 11.388.784 KB Temp: 104 KB Used: 11.388.888 KB Free: 11.139.112 KB

Log: 0 %

Total: 3.173.488 KB Used: 8 KB Free: 3.173.480 KB

Sessions: 2 %

Used: 1 Free: 49

General			
Name	hotlc:HOTSVC	Auto Log	On
Version	7.5.00.10	Command Monitor	Off
Operating System	AIX (UNIX)	Resource Monitor	Off
Rundirectory	/sapdb/data/wrk/HOTSVC	OMS Memory Analysis	Off
Start On	3/2/2004 11:22:52 AM	Database Trace	Off

Data Cache	
Total	1.993.216 KB
Used	704 KB
Hit Rate	100 %

Hot Standby			
▷ Official Node	HOTLC		
Node #1	ENHOT1	Master	ONLINE
Node #2	ENHOT2	Standby	STANDBY

hotlc:HOTSVC

Figure 19: DBMGUI / Main screen

4 Storage Systems

The API is available for the IBM TotalStorage™ Enterprise Storage Server (IBM 2105), IBM TotalStorage™ DS8000 (IBM 2107) and IBM TotalStorage™ SAN Volume Controller (IBM 2145).
Please visit the IBM WEB pages for all IBM TotalStorage™ products to learn more about the products http://www.storage.ibm.com/.

4.1 Enterprise Storage Server

The IBM TotalStorage™ Enterprise Storage Server (ESS) Model 800 helps set new standards in performance, automation and integration as well as capabilities that support continuous availability to data for the on demand world. This storage system also supports many advanced functions, which can be critical for increasing data availability during planned outages and for protecting data from planned and unplanned outages. These advanced functions can provide important disaster recovery and backup protection.
The homepage http://www.storage.ibm.com/disk/ess/index.html will guide you to all product specific documentations, services, upgrades etc.

To configure the ESS you could either use the 'ESS specialist' or the 'Command Line Interface' (CLI). The documentation could be found at the above www link.

4.1.1 Installing the ESScli

Install the correct Command Line Interface on all Nodes of the Hot Standby System, regarding the installation instruction.

4.1.1.1 Changes to the ESScli

The storage specific library 'libHSSibm2105.so' uses functions of the ESS Command Line Interface (CLI). The standard installation has the user rights of

```
# ls -l
dr-xr-xr-x   3 root     system          512 Dec 31 1969  ibm2105cli
# cd ibm2105cli
# ls -l
-r-xr-xr-x   1 root     system         1801 Jun 06 18:58 CLI.CFG
-rwxr-x--x   1 root     system         7277 Jun 06 18:59 README_AIX.txt
dr-xr-x---   4 root     system          512 Dec 31 1969  jre
-r-xr-x--x   1 root     system       920536 Jun 06 18:56 rsCopyServices.jar
-r-xr-x--x   1 root     system         9854 Jun 06 18:58 rsExecuteTask.sh
-r-xr-x--x   1 root     system         8289 Jun 06 18:58 rsList2105DPO.sh
-r-xr-x--x   1 root     system        12766 Jun 06 19:00 rsList2105s.sh
-r-xr-x--x   1 root     system        10568 Jun 06 18:59 rsPrimeServer.sh
-r-xr-x--x   1 root     system        11333 Jun 06 18:56 rsQuery.sh
-r-xr-x--x   1 root     system        10455 Jun 06 18:58 rsQueryComplete.sh
-r-xr-x--x   1 root     system         5292 Jun 06 18:56 rsTestConnection.sh
```

To get the library up and running the database special user sdb in group sdba needs to run java applications. Therefore he needs to be a member of the system group or the user rights needs to be changed for the directory `ibm2105cli`.

Change the user rights for all `*.jar` and `*.sh` from 551 to 555 and for the directory `jre` from 550 to 555.

Example:

```
# chmod -R 555 /usr/opt/ibm/ibm2105cli/jre
# chmod 555 /usr/opt/ibm/ibm2105cli/*.jar
# chmod 555 /usr/opt/ibm/ibm2105cli/*.sh
```

4.1.2 Create Copy Server tasks

Use the WEB GUI to create the copy server tasks. For a complete documentation refer to **Redbook 'IBM TotalStorage Enterprise Storage Server, Implementing Copy Services in an Open Environment'**.

Create full copy flashCopies (not persistent).

4.2 DS 8000

The DS8000 series is designed to provide exceptional performance while adding virtualization capabilities that can help you allocate system resources more effectively and better control application quality of service. The DS8000 series also offers powerful functions that are designed to help protect data from unforeseen events and maintain data availability, which can benefit businesses that must have round the clock access to information.

The homepage http://www-03.ibm.com/servers/storage/disk/ds8000/index.html will guide you to all product specific documentations, services, upgrades etc.

To configure the DS you could either use the 'DS GUI' or the 'DS command line interface' (DScli). The documentation could be found at the above www link.

4.2.1 Installing the DScli

The DScli needs to be installed on all HSS nodes. Please make sure that the correct version is installed to support the HSS implementation for the DS 8000 series.

4.2.1.1 Changes to the DScli

To get the library up and running the database special user sdb in group sdba needs to run java applications. Therefore the user needs to be a member of the `system` group or the user rights needs to be changed for the home directory `/opt/ibm` and `/opt/ibm/DScli`.

Change the user rights to 555.

Example:

```
# chmod -R 555 /opt/ibm
# chmod -R 555 /opt/ibm/DScli
```

4.3 SAN Volume Controller

IBM TotalStorage™ SAN Volume Controller is designed to reduce the complexity and costs of managing storage networks. It allows users to virtualizes their storage and helps increase the utilization of existing capacity and centralize the management of multiple controllers in an

open-system SAN environment. The SAN Volume Controller now supports attachment to non-IBM storage systems. Now storage administrators can reallocate and scale storage capacity and make changes to more underlying storage systems without disrupting applications.
At http://www.storage.ibm.com/software/virtualization/svc/index.html you will get all information you need about the SVC (e.g. supported hardware list, services, whitepapers, planning guide).

To configure the SVC you could use the GUI or the 'command line interface' (cli). The SVCCLI requires a Secure Shell (SSH) connection to the SVC. Since the API is using SVCCLI commands it is required to install the SSH for the owner of the database. Refer to the SVC documentation to learn how to set up SSH. The Redbook

- IBM TotalStorage SAN Volume Controller (SG24-6423-02) at http://www.redbooks.ibm.com/redbooks/pdfs/sg246423.pdf

This Redbook includes a detailed explanation for Windows® and AIX.
Additional information how to install SSH on AIX systems could be found in following Redbooks:

- Managing AIX Server Farms (SG24-6606-00)
- An Introduction to Security in a CSM 1.3 for AIX 5L Environment (SG24-6873-00)

5 Database System

All administration commands on the master and the standby instance are called on the master instance. This chapter will provide information about Takeover, Backup/Recovery, Checks and Parameters.

5.1 Takeover

HACMP observes the running master instance. It performs a takeover when the master database fails. The dbmcli-command db_online performs a takeover for database instances in state Standby.

```
$dbmcli –n <official node> –d <db_name> –u <dbm-user>,<password> db_online
```

The previously failed master instance can be activated as standby when the server and the database volumes are available. Use the Configuration Wizard of the DBMGUI or the dbmcli commands.

```
$dbmcli –n <official node> –d <db_name> –u <dbm-user>,<password>
$ dbmcli on <official node>: <dbname> > hss_execute <standby> db_offline
$ dbmcli on <official node>: <dbname> > db_standby <standby>
```

5.2 Planned Shutdown and Restart of the Master

The standby instance doesn't shutdown with a shutdown of the master. Reactivate the standby instance after a shutdown of the master.

```
$dbmcli –n <official node> –d <db_name> –u <dbm-user>,<password>
$ dbmcli on <official node>: <dbname> > db_offline
$ dbmcli on <official node>: <dbname> > hss_execute <standby> db_offline
…
$ dbmcli on <official node>: <dbname> > db_online
$ dbmcli on <official node>: <dbname> > db_standby <standby>
```

5.3 Check the Master Configuration

System tables show the status of the master and the standby.

```
$sqlcli –n <official node> –d <db_name> –u <dba-user>,<password>
$ sqlcli on <official node>: <dbname> > select * from hoststandbycomponent
$ sqlcli on <official node>: <dbname> > select * from hoststandbygroup
```

5.4 The Standby Instance is Read Only

The standby instance supports reads from the database. Modifications on the data in the standby instance are not allowed.

```
$sqlcli –n <standby> –d <db_name> –u <dba-user>,<password>
$ sqlcli on <standby>: <dbname> > select * from …
```

5.5 Parameter Changes

Change parameter values on the master instance only. The command db_standby copies all parameters from the master to the standby instance. This is one reason why the data- and log volumes have the same names and sizes with the master and the standby instance.

Parameter Changes become valid with a restart of the database. Use the command db_standby to activate the standby instance after a shutdown of the master instance (see chapter 5.2 Planned Shutdown and Restart of the Master).

5.6 Verify

Administrators can perform a verify on the master and on the standby instance.

```
$dbmcli –n < official node > -d <db_name> –u <dbm-user>,<password>
$ dbmcli on <official node>: <dbname> > hss_execute sql_execute check data
```

The verify checks the logical consistency of the database. It reads all occupied data blocks and calculates a checksum and compares it with the checksum written to the block.
The master and standby write all modified data blocks into the own data area. The check on the master can report corrupted data blocks, which are not corrupted, on the standby (visa versa).

5.7 Add Data Volume

Add the new data volume to the master instance in state Admin or Online. The standby can continue. It can run into a database full situation until the volume is available for the standby instance as well. The command db_standby doesn't work until the next restart of the master instance (see chapter 5.2).

Additional steps might be required on the storage subsystem.
ESS: use consistency groups if the data volumes are spread over more than one ESS volume
DS8000: use comma separated list for the DS volume_ID in the RTEHSS_config file
SVC: use a comma separated list for the VDiskID in the RTEHSS_config file

5.8 Backup/Restore

The database works with the defined log area consisting of the log volumes. It overwrites pages in the log volumes if the pages are saved. Use the autosave log feature and let the database kernel create the log backups.
The master database can overwrite log pages if they are saved. The standby instance stops, if it can't find the needed log pages in the log volumes anymore. This can happen, if the standby instance keeps offline as the master is online and produces log entries. The next activation of the standby instance would lead to an init standby. The master stays online, even the standby is not able to read and apply log entries fast enough. Use a large log area to prevent from this situation.

Master and Standby work with one Shared Log; i.e. there is one physical log history for both instances. Each instance has it's own log history protocol file dbm.knl. The file dbm.knl is placed in the Rundirectory of the database instance. The Rundirectory is not shared.

Create backups for log and data only on the master instance. The command db_standby copies the file dbm.knl from the master to the standby. Create a medium with the same definition for log backups on the master and the standby. The backup medium should point to a file system that is taken over with a HACMP failover / fall back procedure.

The backups can be restored on all nodes regardless which server created it. Incremental backups are not supported in a hot standby environment.

6 High availability software

6.1 Preparation of the host system

In case of flash copy the vg hosting the log volume needs to be accessible from both hosts. Therefore use ‘varyonvg –u *vgname*’. If you use the autovaryonvg for the logvg, the ‘-u’ option will not be used. In case of a failover/fallback caused by a server outage and the following start-up process, it is required to run the ‘varyonvg’ command for the logvg either with the HACMP or appropriate start-up script.

7 HACMP Environment and Layout

This chapter describes the cluster landscape and environment as we used for testing the solution, instancing a possible productive system configuration.

7.1 HACMP System Landscape

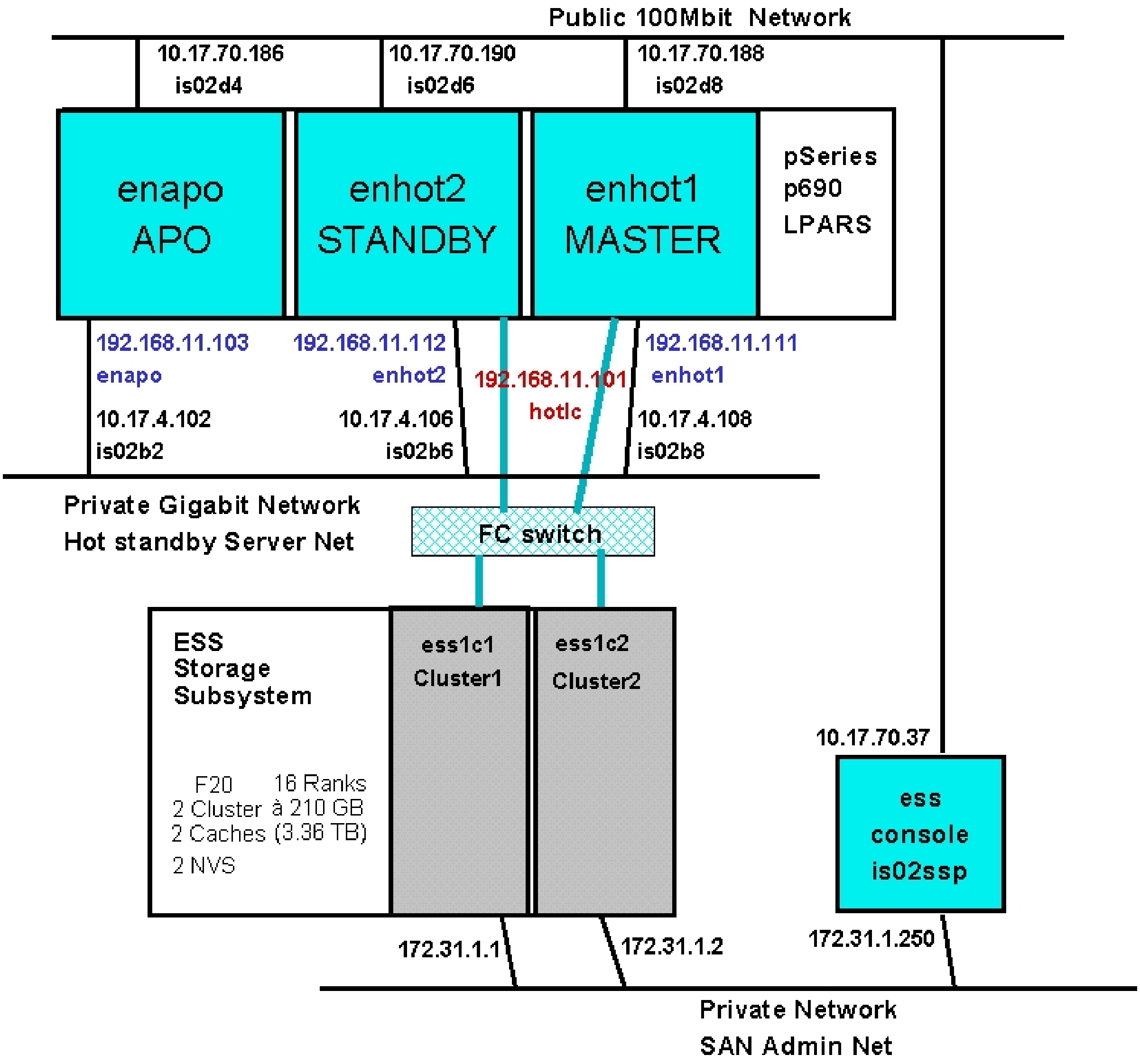


Figure 20: System and Server Landscape

7.2 HACMP Network Layout and Configuration

- Sample network configuration (with only one adapter for service)

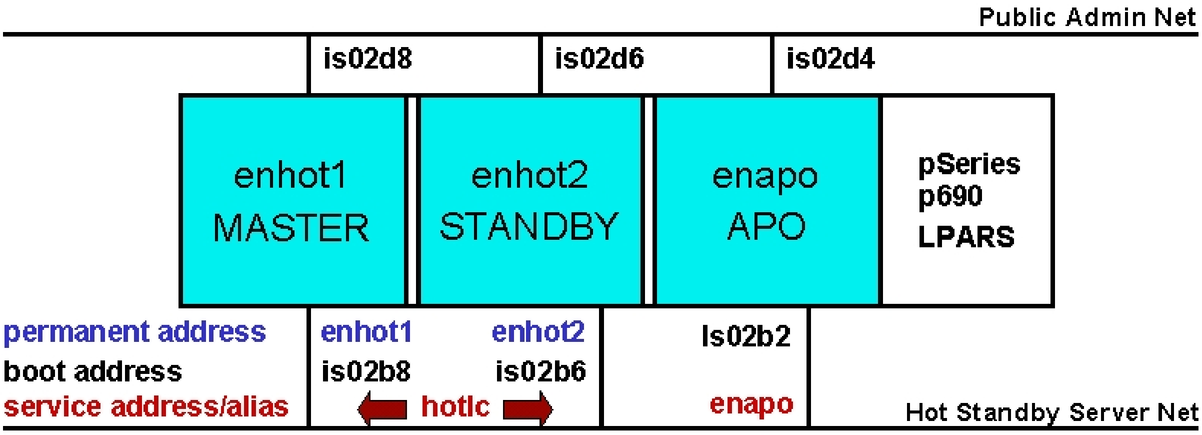


Figure 21: Network and IP-Configuration

<u>Attention:</u>

That's just an example, simplified for testing.

It's required and highly recommended to have at least two adapter at the network used for service (with boot-addresses configured) to ensure high availability at network level eliminating any single point of failure!

- Service IP-address takeover (here: "hotlc") controlled and managed by HACMP

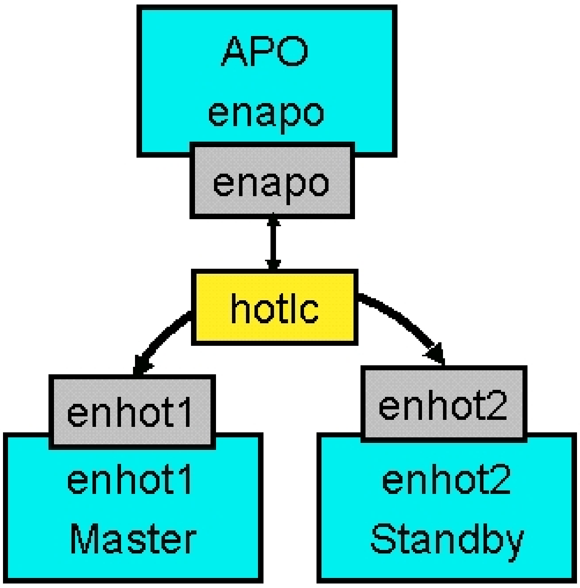


Figure 22: Service IP-Addresses and Takeover (IPAT)

7.3 HACMP Disk and Storage Layout

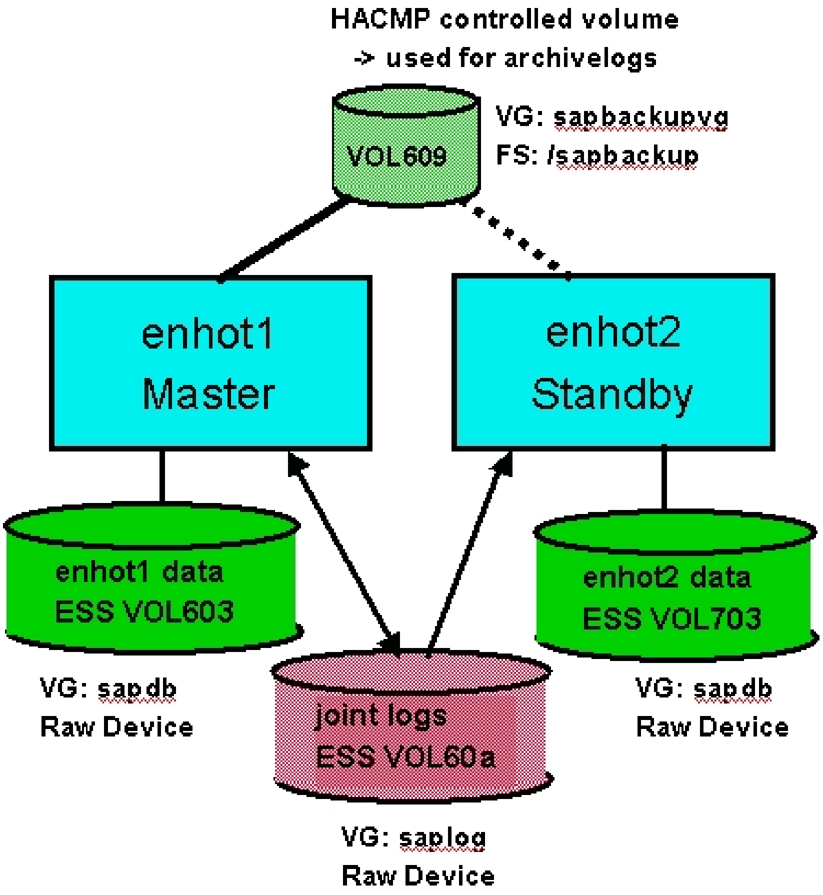


Figure 23: Disks, Volumegroups and Filesystems

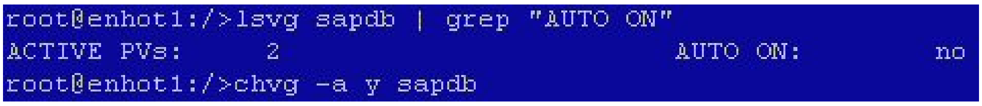
8 Setting up HACMP

8.1 Prepare Environment for HA

8.1.1 Volumegroups

Check and potentially set Volumegroups with local data to varyon automatically.

Example:

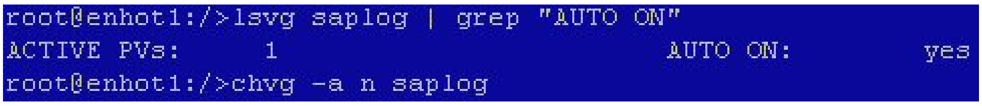


```
root@enhot1:/>lsvg sapdb | grep "AUTO ON"
ACTIVE PVs:      2                           AUTO ON:         no
root@enhot1:/>chvg -a y sapdb
```

Remark: Should be checked and possibly changed on both nodes

- Check and potentially set Volumegroups with shared data to not varyon automatically => that will be done by HACMP (Applicationserver script).

  Example:



```
root@enhot1:/>lsvg saplog | grep "AUTO ON"
ACTIVE PVs:      1                           AUTO ON:         yes
root@enhot1:/>chvg -a n saplog
```

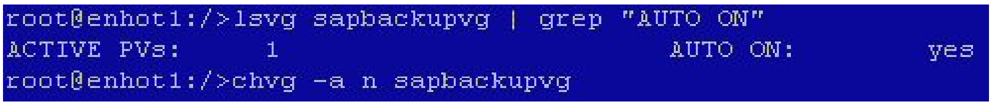
Remark: Make that change on both participating nodes

Attention:

If that environment should be handled at commandline level (without using HACMP), you have to mount that special Volumegroup with command option "-u" on both nodes, at master and at hot-standby server.

- Check and potentially set Volumegroups controlled and taken over by HACMP to not varyon automatically.

  Example:



```
root@enhot1:/>lsvg sapbackupvg | grep "AUTO ON"
ACTIVE PVs:      1                           AUTO ON:         yes
root@enhot1:/>chvg -a n sapbackupvg
```

Remark: Make that change on both participating nodes

8.1.2 Network and Interfaces

With present Hot-standby solution it makes sense and we recommend to use IP-address setup and IPAT (IP Address Takeover) via IP-aliasing. The reason is that with this the node-bound IP- address (persistent address), used for the Hot-standby solution by the DB-server nodes among themselves, can be made high available and the time of a takeover is even faster compared to traditional IP-address configuration.

While IP-aliasing is not supported in HACMP until Version 4.5 the consequence is that you have to use at least HACMP Version 4.5 or higher.

For HACMP it's highly recommended to have at least two interfaces for the service IP-address available (refer to the HACMP 5.1 Administration Guide).

- Configure boot addresses on all interfaces, any of those defined in different subnet.

- Prepare a persistent IP-address ("/etc/hosts" and/or DNS) equaling the hostname for each node. That "permanent" address will be setup in HACMP later, but it will stay as IP-alias on that particular node afterwards even without HACMP activated (so that the node can be reached via that address as long as the system is alive).
- Prepare a "service" IP-address ("/etc/hosts" and/or DNS) to be taken over by HACMP. That IP-address is an IP-alias either, and it's used to connect any client to the Master DB whatever node just being the "Master".

8.2 HACMP Configuration and Samples

The following setup and sample configuration is based on HACMP 5.1 just as used in our test environment. Using different HACMP Version(s) may differ in the way the configuration is done and the example screenshots below will look different as well, but at last the content, the environment and the setup should be the same.

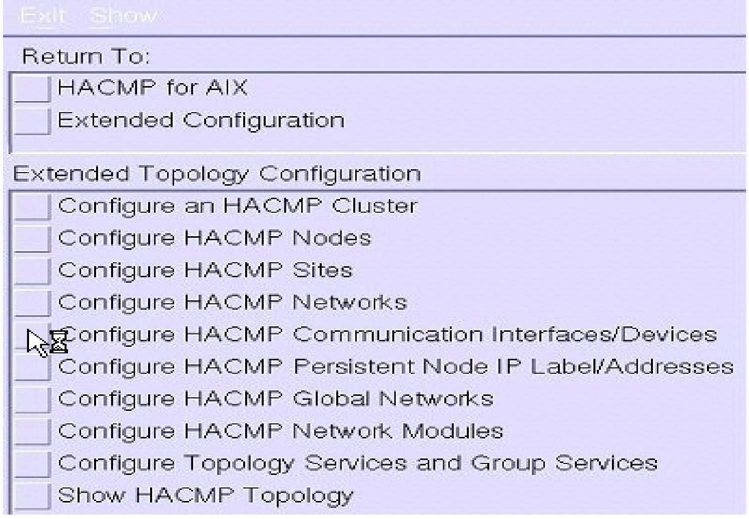
8.2.1 Initialization and Standard Configuration

Following up the HACMP 5.1 Administration Guide define and configure

- HACMP Cluster
- Nodes belonging to that cluster
- Initial IP-address for each node

and synchronize that HACMP configuration to distribute the setup to all/both cluster nodes.

8.2.2 Extended Topology Configuration



Exit Show

Return To:

□ HACMP for AIX

□ Extended Configuration

Extended Topology Configuration

□ Configure an HACMP Cluster

□ Configure HACMP Nodes

□ Configure HACMP Sites

□ Configure HACMP Networks

□ Configure HACMP Communication Interfaces/Devices

□ Configure HACMP Persistent Node IP Label/Addresses

□ Configure HACMP Global Networks

□ Configure HACMP Network Modules

□ Configure Topology Services and Group Services

□ Show HACMP Topology

Figure 24: HACMP Topology Configuration

- Select submenu **Configure HACMP Communication Interfaces/Devices** and add all your additional (Boot) interfaces and IP-addresses
- Select submenu **Configure HACMP Persistent Node IP Label/Addresses** and configure node-bound service IP-addresses as prepared before.

8.2.3 HACMP Extended Resources Configuration

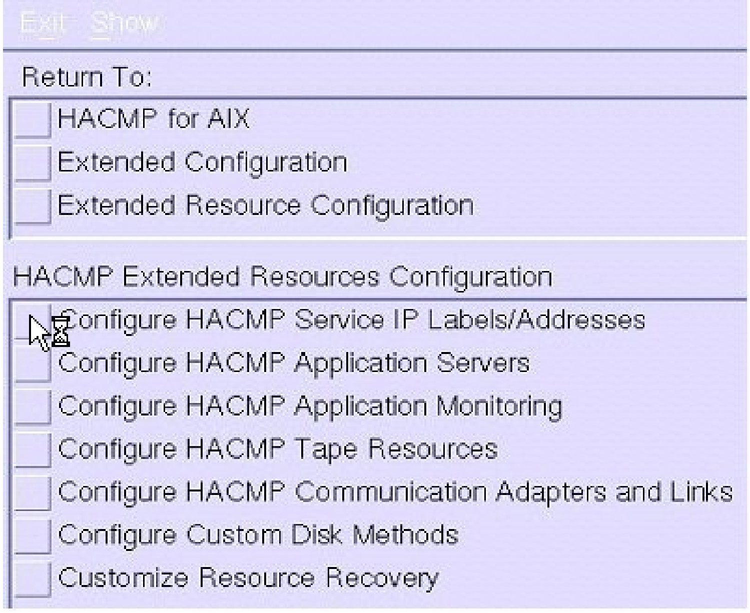
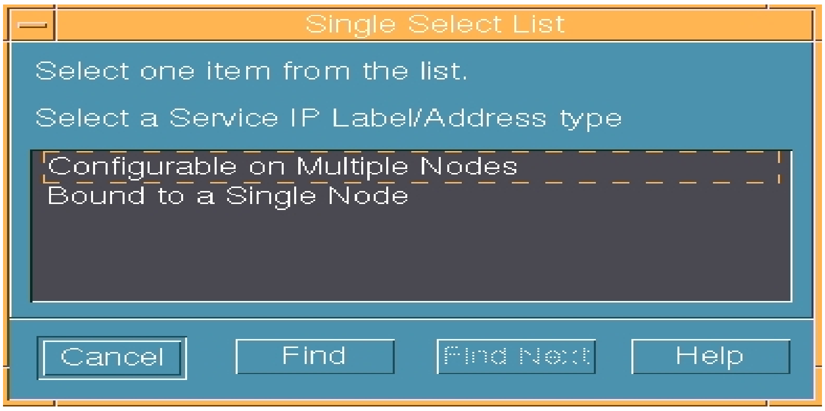

Exit Show
Return To:
HACMP for AIX
Extended Configuration
Extended Resource Configuration
HACMP Extended Resources Configuration
Configure HACMP Service IP Labels/Addresses
Configure HACMP Application Servers
Configure HACMP Application Monitoring
Configure HACMP Tape Resources
Configure HACMP Communication Adapters and Links
Configure Custom Disk Methods
Customize Resource Recovery

Figure 25: HACMP Extended Resources Configuration

a) Service IP-address

- Select submenu **Configure HACMP Service IP Labels/Addresses** and configure node-bound service IP-addresses as prepared before. Choose "Configurable on Multiple Nodes" in the popup window appearing:



Select one item from the list.

Select a Service IP Label/Address type

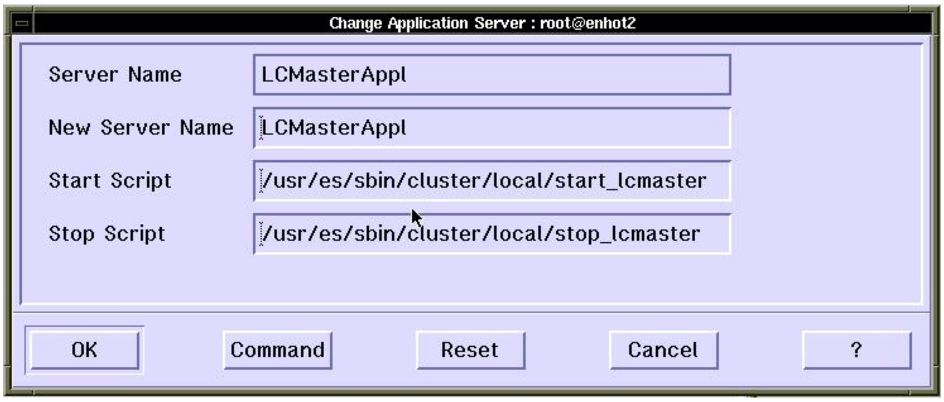
Figure 26: HACMP Service IP Label/Address

and setup these service IP-address which should be taken over virtually, bound to the Master Database, used to connect from any client to that DB.

b) HACMP Application Server

- Select submenu **Configure HACMP Application Servers** and define two HACMP Application Server – one managing the Master DB (we called it "LCMasterAPPL") and the other one handling the Hot-standby server (that's called "LCHotAppl" by us).

Enter scripts "start_lcmaster" and "stop_lcmaster" defining the Master:

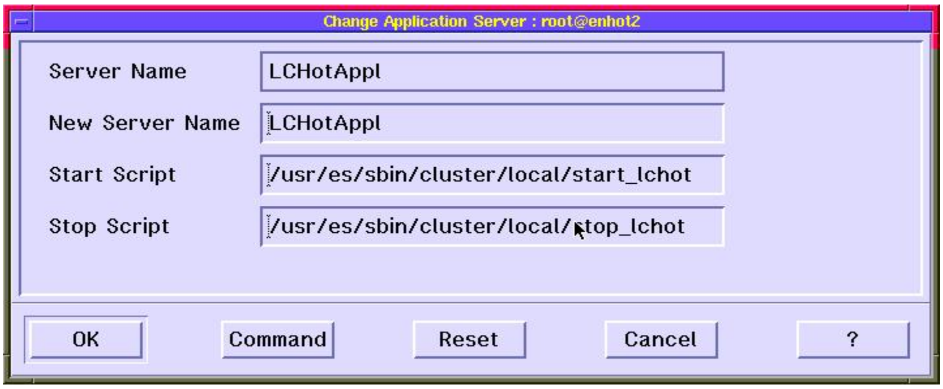


Change Application Server : root@enhot2

Server Name: LCMasterAppl
New Server Name: LCMasterAppl
Start Script: /usr/es/sbin/cluster/local/start_lcmaster
Stop Script: /usr/es/sbin/cluster/local/stop_lcmaster

Figure 27: HACMP Application Server - Example for Master Server

Enter scripts “start_lchot” and “stop_lchot” defining the Standby:



Change Application Server : root@enhot2

Server Name: LCHotAppl
New Server Name: LCHotAppl
Start Script: /usr/es/sbin/cluster/local/start_lchot
Stop Script: /usr/es/sbin/cluster/local/stop_lchot

Figure 28: HACMP Application Server - Example for Hot-Standby

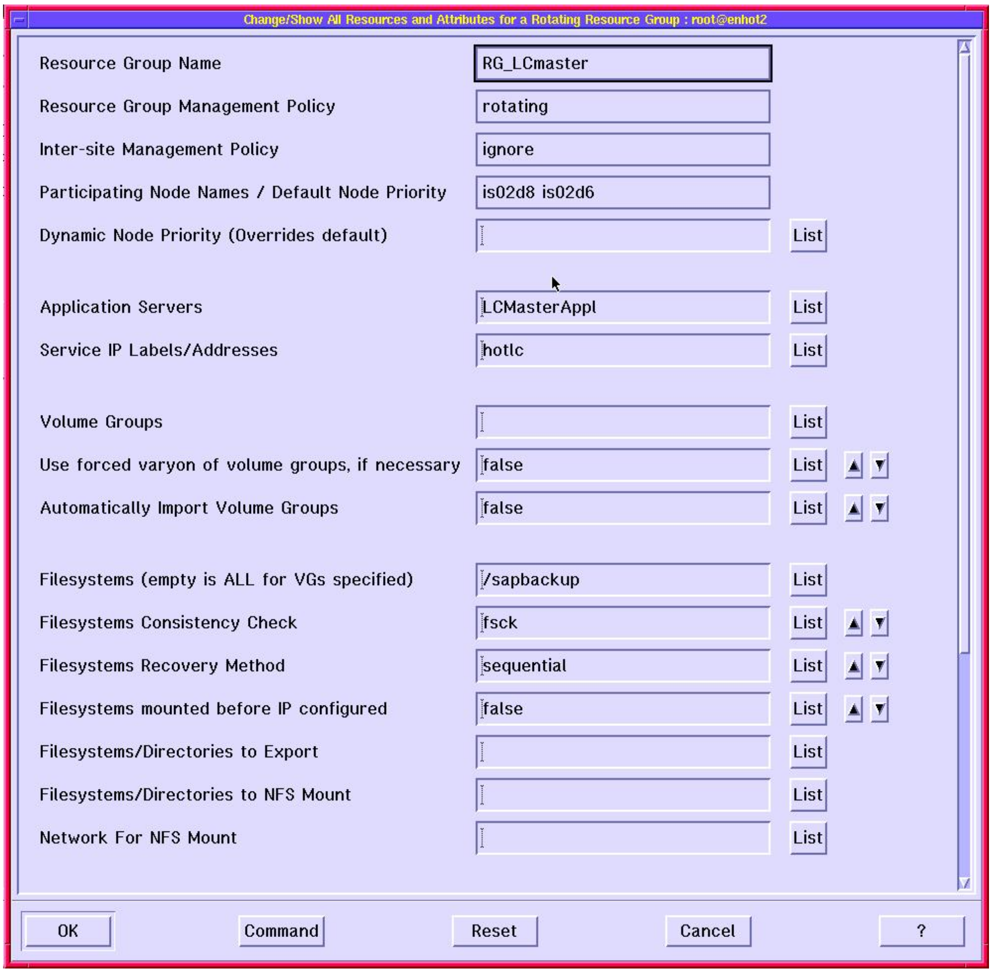
8.2.4 HACMP Extended Resource Group Configuration

- Select submenu **Add a Resource Group** and define three Resource Groups – one managing the Master Database and both others handling the Hot-standby server.
  ➢ Define the first Group used for Master DB with “Rotating” resources;
  we named it “RG_LCmaster”
  ➢ Define both others used for Hot-standby with “Cascading” resources;
  we named them “RG_LChot1” and “RG_LChot2”

The Function of the “Master Resource Group” first is simply to watch the Master DB server and takeover all defined resources to the standby node – to takeover shared service IP-address, shared file system(s) and to restart the Master DB, respectively moves Standby DB to Master Database by setting the Hot-Standby database “ONLINE”.
After that the underlying start-script also handles (tries to restart) the Standby DB at the remote/standby node – if actually possible.
In case the master server fails, the standby node takes over and activates the Master DB by changing it’s Standby DB to master functionality. But now the remote node is down (cause it failed before), that start-script is not able to restart the Standby DB.
For that reintegration situation, the standby node – whichever it is – needs a own resource group definition (one for each particular node) with a belonging HACMP application server and a assigned start-script as well (as defined before), but no other resources defined.

- Select submenu **Change/Show Resources and Attributes for a Resource Group** select the just created “Master Resource Group”
  - ➢ Choose both nodes to participate in that group
  - ➢ Choose the just defined “Master Application Server”
  - ➢ Choose the Service IP Address/Label, also just defined to be taken over
  - ➢ Choose the volume group defined before to be taken over, or – as we used to setup – directly choose the affected filesystem(s) to be taken over

Example:



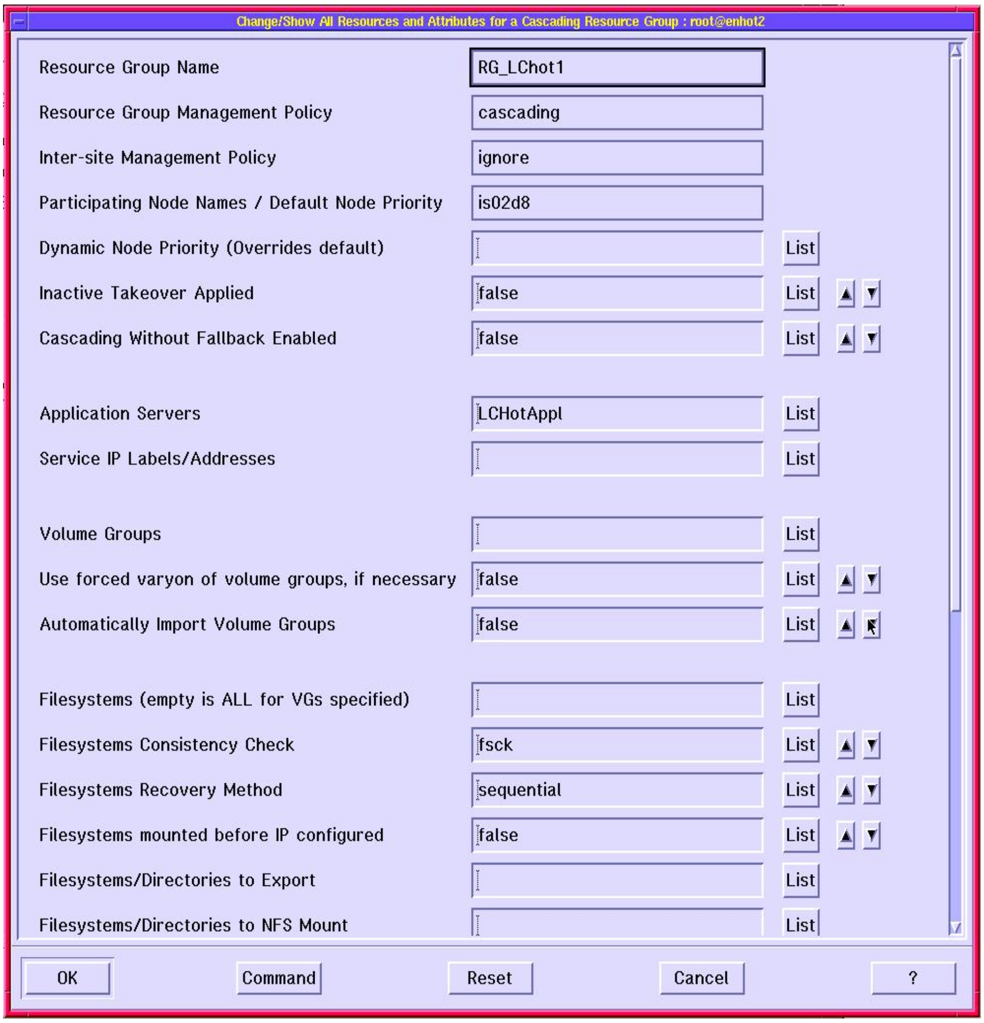
Change/Show All Resources and Attributes for a Rotating Resource Group : root@enhot2

Resource Group Name	RG_LCmaster
Resource Group Management Policy	rotating
Inter-site Management Policy	ignore
Participating Node Names / Default Node Priority	is02d8 is02d6
Dynamic Node Priority (Overrides default)	
Application Servers	LCMasterAppl
Service IP Labels/Addresses	hotlc
Volume Groups	
Use forced varyon of volume groups, if necessary	false
Automatically Import Volume Groups	false
Filesystems (empty is ALL for VGs specified)	/sapbackup
Filesystems Consistency Check	fsck
Filesystems Recovery Method	sequential
Filesystems mounted before IP configured	false
Filesystems/Directories to Export	
Filesystems/Directories to NFS Mount	
Network For NFS Mount	

Figure 29: HACMP Resources - Example for Master Server

- Select submenu **Change/Show Resources and Attributes for a Resource Group** select one of the just created “Standby Resource Groups”
  - ➢ Choose the appropriate node to which that group should belong
  - ➢ Choose the just defined “Standby Application Server”
  - ➢ Do NOT choose a Service IP Address/Label

Example:



Change/Show All Resources and Attributes for a Cascading Resource Group : root@enhot2

Resource Group Name	RG_LChot1
Resource Group Management Policy	cascading
Inter-site Management Policy	ignore
Participating Node Names / Default Node Priority	is02d8
Dynamic Node Priority (Overrides default)	
Inactive Takeover Applied	false
Cascading Without Fallback Enabled	false
Application Servers	LCHotAppl
Service IP Labels/Addresses	
Volume Groups	
Use forced varyon of volume groups, if necessary	false
Automatically Import Volume Groups	false
Filesystems (empty is ALL for VGs specified)	
Filesystems Consistency Check	fsck
Filesystems Recovery Method	sequential
Filesystems mounted before IP configured	false
Filesystems/Directories to Export	
Filesystems/Directories to NFS Mount	

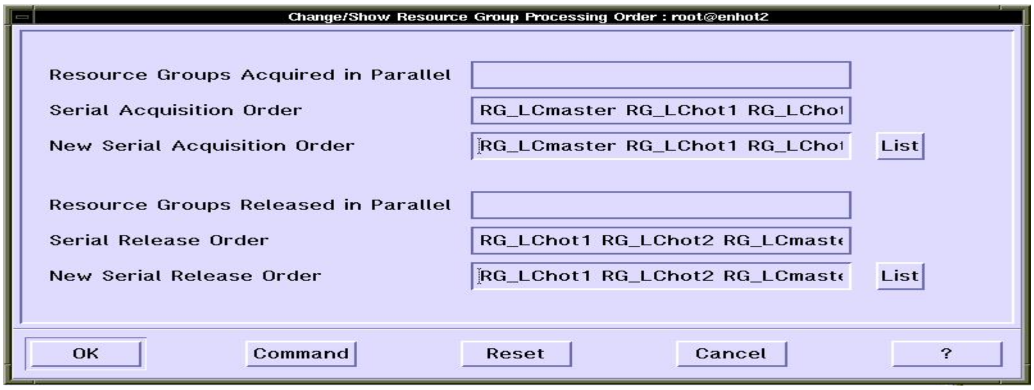
Figure 30: HACMP Resources - Example for Hot-Standby

8.2.5 Configure Resource Group Run-Time Policies

- Select submenu **Configure Resource Group Processing Ordering** and change the “Acquisition Order” from processing *parallel* to *serial* order and assure that the “Master” resource group defined for the Master-DB is the first in place.

The order of “Release Order” doesn’t care, because stopping Hot-Standby is defined without any action, so it makes no difference.

Result:


Change/Show Resource Group Processing Order : root@enhot2

Resource Groups Acquired in Parallel
Serial Acquisition Order: RG_LCmaster RG_LChot1 RG_LChot
New Serial Acquisition Order: RG_LCmaster RG_LChot1 RG_LChot
Resource Groups Released in Parallel
Serial Release Order: RG_LChot1 RG_LChot2 RG_LCmaste
New Serial Release Order: RG_LChot1 RG_LChot2 RG_LCmaste

Figure 31: HACMP Resource Group Processing Order

8.2.6 HACMP Application Monitoring (optional)

- Select submenu **HACMP Extended Resources Configuration** and **Configure HACMP Application Monitoring**, then choose submenu **Configure Custom Application Monitor** and **Add Custom Application Monitor**:

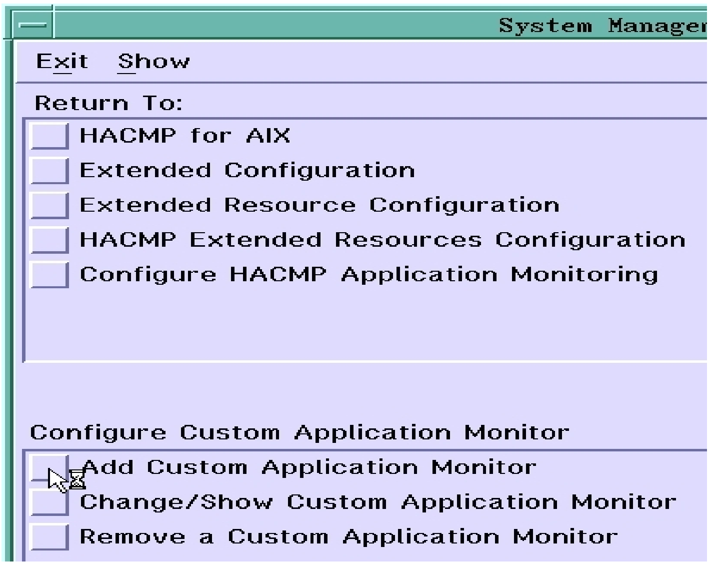
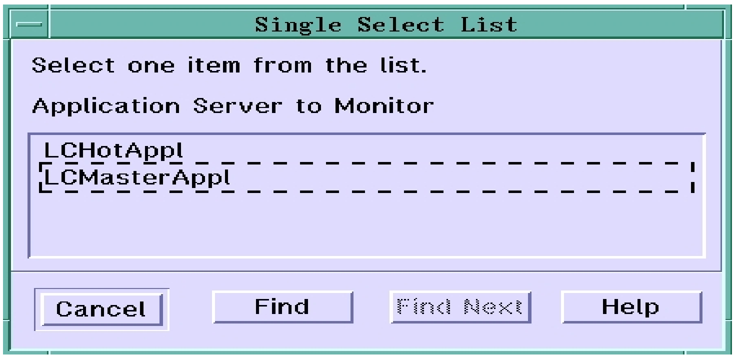

System Manager
Exit
Show
Return To:
HACMP for AIX
Extended Configuration
Extended Resource Configuration
HACMP Extended Resources Configuration
Configure HACMP Application Monitoring
Configure Custom Application Monitor
Add Custom Application Monitor
Change/Show Custom Application Monitor
Remove a Custom Application Monitor

Figure 32: HACMP Application Monitoring Definition

- Choose Application Server for Master (as defined before) in the popup window appearing:


Single Select List

Select one item from the list.

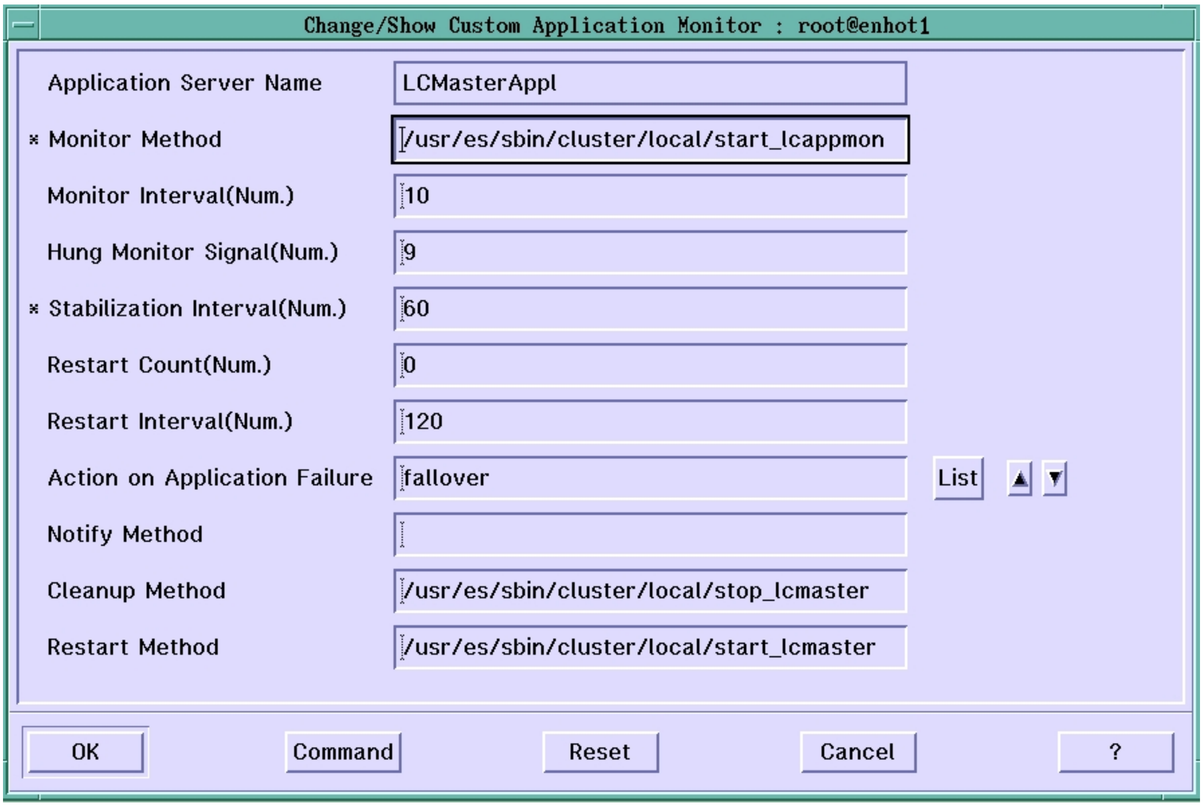
Application Server to Monitor

LCHotAppl
LCMasterAppl

Figure 33: HACMP Application Monitoring – Belonging Application Server

- Enter monitoring Method (own script, see Appendix), time intervals and counts as required
  - ➢ Monitor Method – define monitoring script including the complete path
  - ➢ Monitor Interval – the time in seconds HACMP waits until that script is restarted after it was finished before; we preferred a short interval to assure a short reaction detecting a master database failure, but it's for your own decision.
  - ➢ Restart Count – **must be zero!** to force HACMP to takeover the master DB to the hotstandby node immediately without trying to restart that DB on the same node. That's the basic principle of this solution!
  - ➢ Action on Application Failure – must be "fallover" for the same reasons
  - ➢ Cleanup Method and Restart Method – leave them "as is"

Example:



Change/Show Custom Application Monitor : root@enhot1

Application Server Name	LCMasterAppl
* Monitor Method	/usr/es/sbin/cluster/local/start_lcappmon
Monitor Interval(Num.)	10
Hung Monitor Signal(Num.)	9
* Stabilization Interval(Num.)	60
Restart Count(Num.)	0
Restart Interval(Num.)	120
Action on Application Failure	fallover
Notify Method	
Cleanup Method	/usr/es/sbin/cluster/local/stop_lcmaster
Restart Method	/usr/es/sbin/cluster/local/start_lcmaster

Figure 34: HACMP Application Monitoring – Example Configuration

8.2.7 Script Configuration

The sample scripts, the scripts we used, are all developed to be for universal use and it's almost not necessary to change or configure them. Nevertheless it's at one's own discretion to change or customize them anytime.

The configuration and customization of the different environments is usually done by setting appropriate parameters in "Part 1" of that profile named "hacmpr3.profile" and located at the same place, in the same path as the sample scripts itself (see Appendix 9.2.1).

8.3 HACMP Flowcharts

8.3.1 Starting Master Database

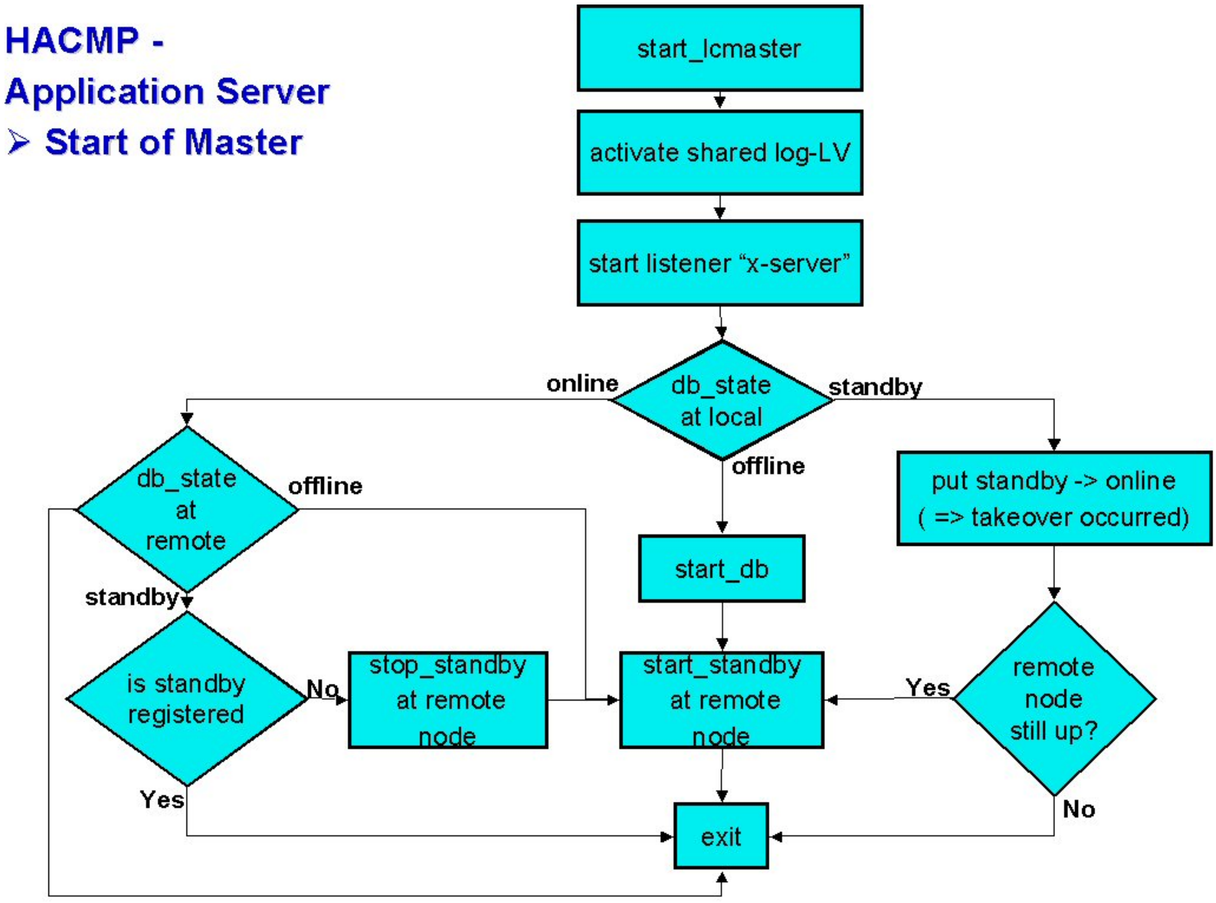


Figure 35: Flowchart HACMP Application Server – Master Server

8.3.2 Starting Hot-Standby Database

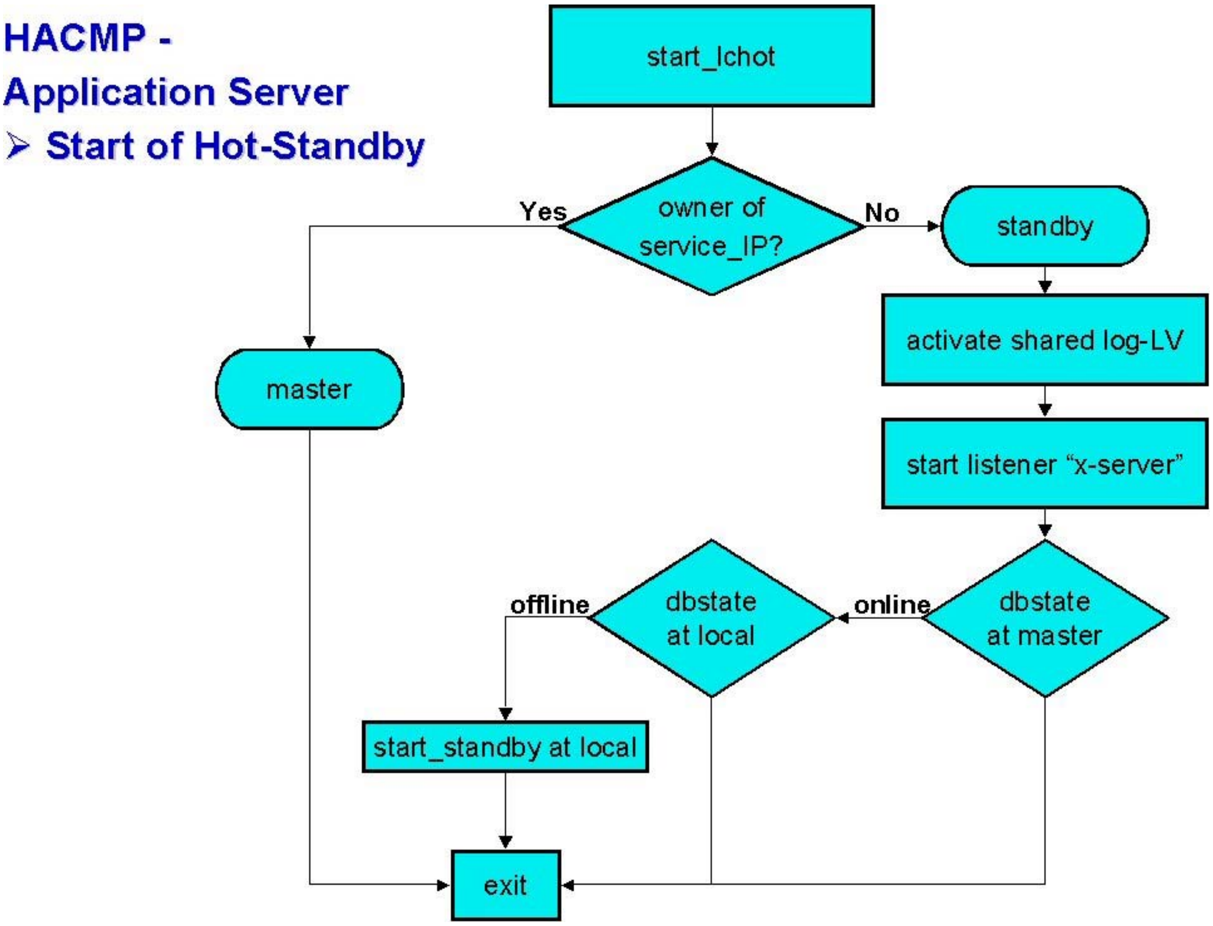


Figure 36: Flowchart HACMP Application Server - Hot-Standby

8.3.3 Stopping Databases

No flowcharts will be shown up here – it's simply not necessary:

- The Master DB is just stopped – it's a single command line entry (see Appendix 9.2.3)
- The Standby DB is not stopped anyway – it's an empty script, just available (see Appendix 9.2.6)

8.3.4 Monitoring Status of Master Database

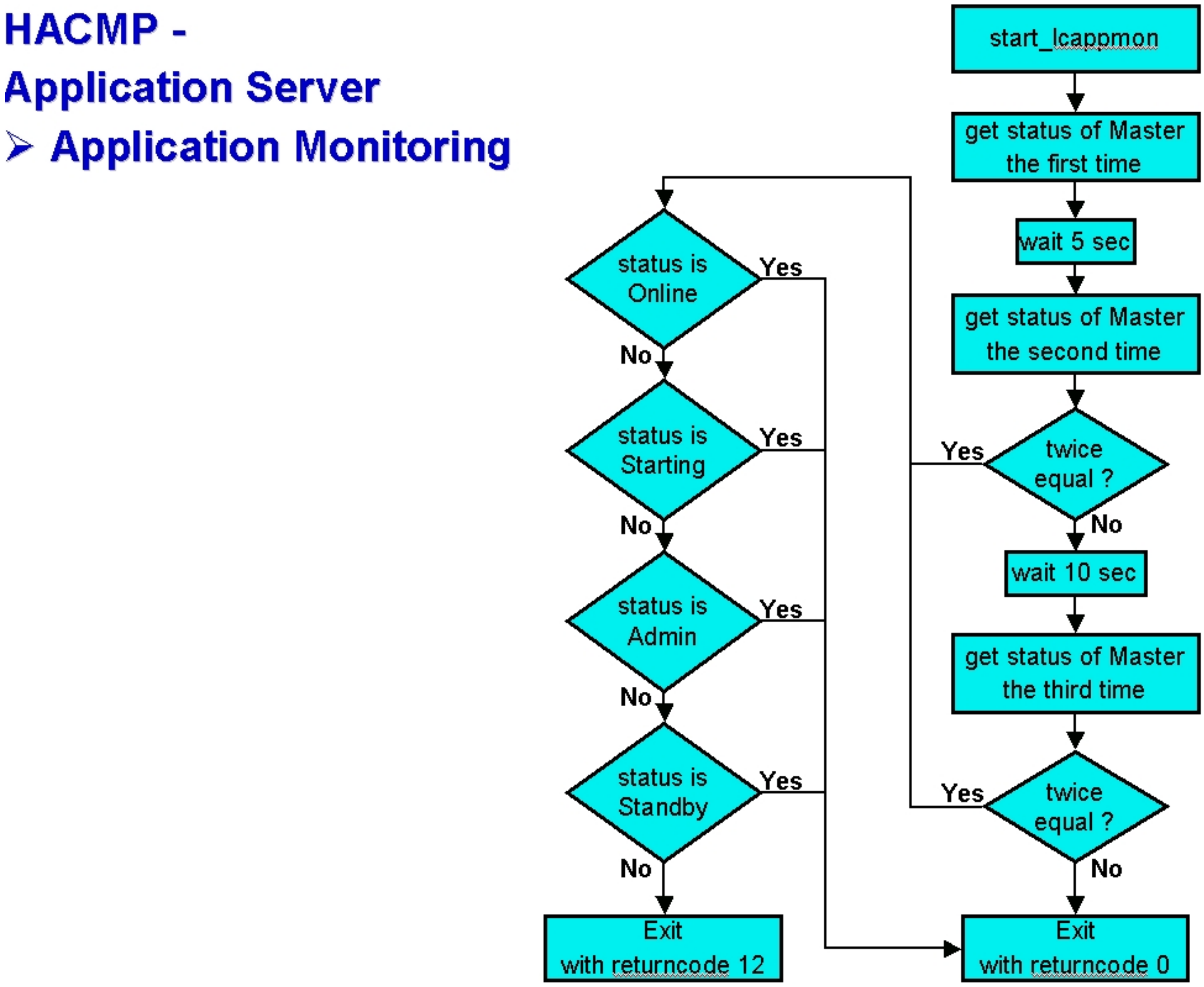


Figure 37: Flowchart HACMP Application Server – Application Monitoring

9 Appendix

9.1 REHSS_config.txt

```
########## RTEHSS_config.txt ##########
# SAP liveCache / MySQL MaxDB
# RunTimeEnvironment HotStandbyStorage configuration file
#
# date                 who            comment
# 05/19/2003   Oliver Goos           initial creation
# 08/01/2005   Oliver Goos           add variable DSdevID to support DS8000
#######################################

# Copy Server Services
# choose either FC or PPRC
CSmode FC
#CSmode PPRC

# OS specific install path for ibm2105cli, or dscli
Ibmclidir /opt/ibm/dscli
# HomeDir of SAP live cache utils
Ibmsapapodir /usr/opt/ibm/ibmsap

# Master liveCache Server
## Case of SVC vdisk_id or vdisk_name
## case of DS disk ID
MlCLogVdiskID 1400
MlCDataVdiskID 1401

# 1st Standby liveCache Server
## Case of SVC vdisk_id
## case of DS disk ID
SlCLogVdiskID 1400
SlCDataVdiskID 1402

# 2nd Standby liveCache Server
## Case of SVC vdisk_id
## case of DS disk ID
SSlCLogVdiskID
SSlCDataVdiskID

# ESS Copy Server/SVC/hmc1
## IP adress
CSaIP 9.155.62.98
## User ID (admin)
CSaUID TIC-TEAM
## User password
CSapwd tic02ds8k

# ESS Copy Server backup/SVC2 / hmc2
## IP adress
CSbIP

# DS storage dev
DSdevID IBM.2107-7573041

# list all HSS_NODE_00x in this section (max. 9)
HSS_NODE_001 p520_TIC3
HSS_NODE_002 p520_TIC4
HSS_NODE_003

# copy server tasks
# ESS: specify task name which was defined on ESS CopyServer to copy data volume from current
MASTER (HS_NODE_00x) to requesting STANDBY (HS_NODE_00y)
# SVC: specify task name which will be used to copy data volume from current MASTER
(HS_NODE_00x) to requesting STANDBY (HS_NODE_00y)
# DS:  specify sequence number which will be used to copy data volume from current MASTER
(HS_NODE_00x) to requesting STANDBY (HS_NODE_00y)
EstDataCST_001_002 1020
EstDataCST_001_003
EstDataCST_002_001 2010
EstDataCST_002_003
EstDataCST_003_001
EstDataCST_003_002
```

```
TermDataCST_001_002 1020
TermDataCST_001_003
TermDataCST_002_001 2010
TermDataCST_002_003
TermDataCST_003_001
TermDataCST_003_002

# in case of remote copy / PPRC
# specify task name which is used to copy log volume from current MASTER (HS_NODE_00x) to
requesting STANDBY (HS_NODE_00y)
EstLogCST_001_002
EstLogCST_001_003
EstLogCST_002_001
EstLogCST_002_003
EstLogCST_003_001
EstLogCST_003_002

TermLogCST_001_002
TermLogCST_001_003
TermLogCST_002_001
TermLogCST_002_003
TermLogCST_003_001
TermLogCST_003_002
```

9.2 HACMP Scripts

9.2.1 Profile

```
#!/bin/ksh
#--------------------------------------------------------------
# Filename:    hacmpr3.profile
# Path:        /usr/es/sbin/cluster/local
# Node(s):     All R/3-Server
# Info:        Main profile to setup HACMP and R/3 variables
#--------------------------------------------------------------
# Part 1
# Variables to be setup to meet your HACMP - R/3 installation.
# May be there are some you don't really need !
#---------------------------------------------------------------

# HACMP environment
#==================
#HAPATH=/usr/sbin/cluster     # HACMP
HAPATH=/usr/es/sbin/cluster   # HACMP/ES
TOOLSDIR=/usr/es/sbin/cluster/local   # HACMP Application-Scripts

# APO/LC environment
#===================
LC_name=HOT                   # Set LiveCache Name
LCADM=sapdb                   # Set Admin of LiveCache
LCHOME=/sapdb/HOT/db          # Set Home-Dir of LiveCache
LCBIN=/sapdb/programs/bin     # Set Programs-Dir of LiveCache
LC_LOG_VolumeGroup=saplog     # Set VG-Name of Hot-Standby Logdevice

# Hostinformations
#=================
LCSERVER=enhot1               # Set hostname of LiveCache-Server
TAKEOVER=enhot2               # Set hostname of Takeover-Host

# IP-Label (=> IP-address)
#=========================
NFSSERVER=hotlc               # Set service IP-label of NFS-Server
LCSERVER_IP=hotlc             # Set service IP-label of LiveCache-Server

LCSERVER_PERM_IP=enhot1       # Set permanent IP-label of LiveCache-Master
STANDBY_PERM_IP=enhot2        # Set permanent IP-label of Standby-Server

# Remote shell command "rsh"
# e.g. choose Kerberos version for SP2
#=====================================
RSH=/usr/bin/rsh              # "normal" rsh
#RSH=/usr/bin/ssh             # Secure rsh
#RSH=/usr/lpp/ssp/rcmd/bin/rsh        # Kerberos Version of rsh
```

```
#---------------------------------------------------------------
# Part 2
# Standard variables used in the scripts, but not to be changed.
# Don't forget to control the filenames and their locations.
#---------------------------------------------------------------

# APO/LC environment
#==================
LCLOWID=`echo $LC_name | /usr/bin/tr '[:upper:]' '[:lower:]'`
#LCADM="$LCLOWID"adm
#LCHOME=$(su - $LCADM "-c dbmcli db_enum" | grep fast | awk '{ print $2 }')

# Logfile
#========
OUT=/tmp/hacmp.out
```

9.2.2 Startscript for Master-DB – start_lcmaster

```
#!/bin/ksh
#---------------------------------------------------------------------------
# Filename:    start_lcmaster
# Path:        /usr/es/sbin/cluster/local
# Node(s):     HACMP-Server; Production LiveCache Server
# Info:        Startscript for HACMP-Server
#---------------------------------------------------------------------------
# First set the environment and dump a timestamp
#---------------------------------------------------------------------------

# set logging mode
[[ "$VERBOSE_LOGGING" = high ]] && set -x

. /usr/es/sbin/cluster/local/hacmpr3.profile

echo "start execution of $TOOLSDIR/start_lcmaster" >> $OUT
echo `date` >> $OUT

#---------------------------------------------------------------------------
# Set Remote Server which runs the Hot-Standby Database
#---------------------------------------------------------------------------
if [ `hostname` = $LCSERVER ]
then
   REMOTE_NODE=$STANDBY_PERM_IP
elif [ `hostname` = $TAKEOVER ]
then
   REMOTE_NODE=$LCSERVER_PERM_IP
else
   echo "This script is not intended to run on this node!" >> $OUT
   echo "The Hot-Standby Database should run on HACMP Cluster nodes only!" >> $OUT
   exit 8
fi

#---------------------------------------------------------------------------
# - export all NFS-Filesystems with the right Permissions (/etc/exports)
# - first activate VolumGroup with shared Log-LV (if not already active)
# - next start x_server (if not already running)
# - get status of LiveCache at service/master node
# - if status of LiveCache is "offline" -> it's an initial startup
#   - start the MaxDB listener
#   - start LiveCache
#   - check if standby node is reachable; if yes:
#     - start the MaxDB listener at remote node
#     - stop Hot-Standby just to be clean
#     - start Hot-Standby at remote node
# - if status of LiveCache is "standby" -> a takeover has occured
#   - set Hot-Standby "online" -> move standby to master
#   - check if standby node is down; if not:
#     - start the MaxDB listener at remote node
#     - restart Hot-Standby at remote node
#---------------------------------------------------------------------------
#/usr/sbin/exportfs -a
```

```
lsvg -o | grep $LC_LOG_VolumeGroup >/dev/null 2>&1
if [ $? != 0 ]
then
   /usr/sbin/varyonvg -u $LC_LOG_VolumeGroup >> $OUT
fi

ps -ef | grep vserver | fgrep -v grep >/dev/null 2>&1
if [ $? != 0 ]
then
   /usr/bin/su - $LCADM "-c x_server start" >> $OUT
fi

LCSTATE=$($LCBIN/dbmcli -n $LCSERVER_IP -d $LC_name -u control,control db_state | grep -v OK |
grep -v State)

case $LCSTATE in

  OFFLINE)
      echo "Initial start - start Master LC locally ........." >> $OUT
      $LCBIN/dbmcli -d $LC_name -u control,control db_online >> $OUT
      echo "......... and Hot-Standby at remote node" >> $OUT

      ping -c 1 -w 1 $REMOTE_NODE >/dev/null 2>&1
      if [ $? = 0 ]
      then
         echo "Send request to start x_server to $REMOTE_NODE" >> $OUT
         DOING="echo \"$TOOLSDIR/start_vserver\" | at now"
         $RSH $REMOTE_NODE $DOING

         echo "Send request to cleanup DB-environment to $REMOTE_NODE" >> $OUT
         $LCBIN/dbmcli -d $LC_name -u control,control hss_execute $REMOTE_NODE db_offline >>
$OUT

         echo "Send request to startup Hot-Standby to $REMOTE_NODE" >> $OUT
         $LCBIN/dbmcli -d $LC_name -u control,control db_standby $REMOTE_NODE >> $OUT
      else
         echo "$REMOTE_NODE is not reachable => check and handle Hot-Standby manually" >> $OUT
         echo "$REMOTE_NODE is not reachable => check and handle Hot-Standby manually"
      fi;;

   STANDBY)
      echo "Takeover occured - set Hot-Standby online and make it Master LC" >> $OUT
      $LCBIN/dbmcli -d $LC_name -u control,control db_online >> $OUT

      ping -c 1 -w 1 $REMOTE_NODE >/dev/null 2>&1
      if [ $? = 0 ]
      then
         echo "$REMOTE_NODE seems to be alive ......" >> $OUT
         echo "Send request to start x_server to $REMOTE_NODE" >> $OUT
         DOING="echo \"$TOOLSDIR/start_vserver\" | at now"
         $RSH $REMOTE_NODE $DOING

         echo "Send request to cleanup DB-environment to $REMOTE_NODE" >> $OUT
         $LCBIN/dbmcli -d $LC_name -u control,control hss_execute $REMOTE_NODE db_offline >>
$OUT

         echo "Send request to startup Hot-Standby to $REMOTE_NODE" >> $OUT
         $LCBIN/dbmcli -d $LC_name -u control,control db_standby $REMOTE_NODE >> $OUT
      else
         echo "$REMOTE_NODE is not reachable => check and handle Hot-Standby manually" >> $OUT
         echo "$REMOTE_NODE is not reachable => check and handle Hot-Standby manually"
      fi;;

   ONLINE)
      echo "Master LC is already online - what about Hot-Standby?" >> $OUT

      ping -c 1 -w 1 $REMOTE_NODE >/dev/null 2>&1
      if [ $? = 0 ]
    then
         STDBY_STATE=$($LCBIN/dbmcli -d $LC_name -u control,control hss_execute $REMOTE_NODE
db_state | grep -v OK | grep -v State)

         case $STDBY_STATE in
```

```
            OFFLINE)
               echo "It's offline - start Hot-Standby at remote node $REMOTE_NODE" >> $OUT
               $LCBIN/dbmcli -d $LC_name -u control,control db_standby $REMOTE_NODE >> $OUT;;

            STANDBY)
               echo "It's standby - check if Standby is registered at master DB" >> $OUT
               REMOTE_NODE_UpCase=`echo $REMOTE_NODE | /usr/bin/tr '[:lower:]' '[:upper:]'`
               if  [  $($LCBIN/sqlcli  -d  $LC_name  -u  superdba,admin  -a  select  LOCALNODE  from
HOTSTANDBYGROUP  where  STATE  =  \'WAIT  FOR  SYNCHRONIZE\'  |  grep  -v  "row"  |  cut  -c  3-32)  =
$REMOTE_NODE_UpCase ]
               then
                  echo  "It's  standby  and  registered  -  seems  all  well  up  and  working  =>
exiting" >> $OUT
               else
                  echo "It's standby but not registered => restart Hot-Standby" >> $OUT
                  $LCBIN/dbmcli   -d   $LC_name   -u   control,control   hss_execute   $REMOTE_NODE
db_offline >> $OUT
                  $LCBIN/dbmcli -d $LC_name -u control,control db_standby $REMOTE_NODE >> $OUT
               fi;;

            *)
               echo  "returncode  $LCSTATE  of  standby_state  not  expected  =>  check  and  handle
Hot-Standby manually" >> $OUT
               echo  "returncode  $LCSTATE  of  standby_state  not  expected  =>  check  and  handle
Hot-Standby manually"
               exit 8;;

         esac
      else
         echo "$REMOTE_NODE is not reachable => check and handle Hot-Standby manually" >> $OUT
         echo "$REMOTE_NODE is not reachable => check and handle Hot-Standby manually"
      fi;;

   *)
      echo "returncode $LCSTATE of db_state not expected => check and handle LC manually" >>
$OUT
      echo "returncode $LCSTATE of db_state not expected => check and handle LC manually"
      exit 8;;

esac

exit 0
```

9.2.3 Startscript for (remote start of) Listener “x_server” – start_vserver

```
#!/bin/ksh
#---------------------------------------------------------------------------
# Filename:    start_vserver
# Path:        /usr/es/sbin/cluster/local
# Node(s):     HACMP-Server; Hot-Standby LiveCache Server
# Info:        Startscript for HACMP-Server
#---------------------------------------------------------------------------
# First set the environment and dump a timestamp
#---------------------------------------------------------------------------

# set logging mode
[[ "$VERBOSE_LOGGING" = high ]] && set -x

. /usr/es/sbin/cluster/local/hacmpr3.profile

echo "start execution of $TOOLSDIR/start_vserver" >> $OUT
echo `date` >> $OUT

#---------------------------------------------------------------------------
# - start x_server (if not already runnning)
#---------------------------------------------------------------------------
ps -ef | grep vserver | fgrep -v grep >/dev/null 2>&1
if [ $? != 0 ]
then
   /usr/bin/su - $LCADM "-c x_server start" >> $OUT
fi
```

```
exit 0
```

9.2.4 Stopscript for Master-DB – stop_lcmaster

```
#!/bin/ksh
#-----------------------------------------------------------------------------
# Filename:    stop_lcmaster
# Path:        /usr/es/sbin/cluster/local
# Node(s):     HACMP-Server; Production LiveCache Server
# Info:        Stopscript for HACMP-Server
#-----------------------------------------------------------------------------
# First set the environment and dump a timestamp
#-----------------------------------------------------------------------------

. /usr/es/sbin/cluster/local/hacmpr3.profile

# set logging mode
[[ "$VERBOSE_LOGGING" = high ]] && set -x

echo "start execution of $TOOLSDIR/stop_lcmaster" >> $OUT
echo `date` >> $OUT

#-----------------------------------------------------------------------------
# - stop LiveCache/Master DB
#-----------------------------------------------------------------------------

$LCBIN/dbmcli -d $LC_name -u control,control db_offline >> $OUT

exit 0
```

9.2.5 Startscript for Standby-DB – start_lchot

```
#!/bin/ksh
#-----------------------------------------------------------------------------
# Filename:    start_lchot
# Path:        /usr/es/sbin/cluster/local
# Node(s):     HACMP-Server; Hot-Standby LiveCache Server
# Info:        Startscript for HACMP-Server
#-----------------------------------------------------------------------------
# First set the environment and dump a timestamp
#-----------------------------------------------------------------------------

# set logging mode
[[ "$VERBOSE_LOGGING" = high ]] && set -x

. /usr/es/sbin/cluster/local/hacmpr3.profile

echo "start execution of $TOOLSDIR/start_lchot" >> $OUT
echo `date` >> $OUT

#-----------------------------------------------------------------------------
# Set Remote Server which runs the Hot-Standby Database -> here it's myself
#-----------------------------------------------------------------------------
if [ `hostname` = $LCSERVER ]
then
   REMOTE_NODE=$LCSERVER_PERM_IP
elif [ `hostname` = $TAKEOVER ]
then
   REMOTE_NODE=$STANDBY_PERM_IP
else
   echo "This script is not intended to run on this node!" >> $OUT
   echo "The Hot-Standby Database should run on HACMP Cluster nodes only!" >> $OUT
   exit 8
fi

#-----------------------------------------------------------------------------
# - check if i am master or standby (do i have the service IP-address available)?
# - if i'm standby:
#   - first activate VolumGroup with shared Log-LV (if not already active)
#   - next start x_server (if not already running)
#   - get status of LiveCache at service/master node
#   - if status of LiveCache at master node is "online"
#     and if status of Hot-Standby is "offline"
#     - stop Hot-Standby just to cleanup
#     - start Hot-Standby at local node
#   - else
#     - exit doing nothing, because LiveCache at master must be activated before
#-----------------------------------------------------------------------------
/usr/bin/netstat -i | grep $LCSERVER_IP >/dev/null 2>&1

if [ $? != 0 ]
then

   lsvg -o | grep $LC_LOG_VolumeGroup >/dev/null 2>&1
   if [ $? != 0 ]
   then
      /usr/sbin/varyonvg -u $LC_LOG_VolumeGroup >> $OUT
#      /usr/sbin/varyonvg -b -u $LC_LOG_VolumeGroup >> $OUT
   fi

   ps -ef | grep vserver | fgrep -v grep >/dev/null 2>&1
   if [ $? != 0 ]
   then
      /usr/bin/su - $LCADM "-c x_server start" >> $OUT
   fi

  LCSTATE=$($LCBIN/dbmcli -n $LCSERVER_IP -d $LC_name -u control,control db_state | grep -v OK
| grep -v State)

   case $LCSTATE in

      ONLINE)
         echo "Master is online - check if i have to start Hot-Standby" >> $OUT
         $LCBIN/dbmcli -d $LC_name -u control,control db_state | grep OFFLINE
         if [ $? = 0 ]
         then
            echo "Hot-Standby is offline => start it up" >> $OUT
            $LCBIN/dbmcli -d $LC_name -u control,control db_offline >> $OUT
            $LCBIN/dbmcli   -n   $LCSERVER_IP   -d   $LC_name   -u   control,control   db_standby
$REMOTE_NODE >> $OUT
         fi;;

      *)
         echo "returncode $LCSTATE of db_state not valid to start Hot-Standby" >> $OUT
         echo "automatically => check state and handle Master and Standby manually" >> $OUT
         echo "returncode $LCSTATE of db_state not valid to start Hot-Standby"
         echo "automatically => check state and handle Master and Standby manually"
         exit 4;;

   esac

fi

exit 0
```

9.2.6 Stopscript for Standby-DB – stop_lchot

```
#!/bin/ksh
#-----------------------------------------------------------------------------
# Filename:    stop_lchot
# Path:        /usr/es/sbin/cluster/local
# Node(s):     HACMP-Server; Hot-Standby LiveCache Server
# Info:        Stopscript for HACMP-Server
#-----------------------------------------------------------------------------
# First set the environment and dump a timestamp
#-----------------------------------------------------------------------------

. /usr/es/sbin/cluster/local/hacmpr3.profile

# set logging mode
[[ "$VERBOSE_LOGGING" = high ]] && set -x

echo "start execution of $TOOLSDIR/stop_lchot" >> $OUT
echo `date` >> $OUT

#-----------------------------------------------------------------------------
```

```
# - don't stop Hot-Standby
#---------------------------------------------------------------------------

echo "This script does nothing:" >> $OUT
echo "The Hot-Standby is not stopped - it will stay up and alive! " >> $OUT

exit 0
```

9.2.7 Monitoringscript checking Status of Master-DB – start_lchappmon

```
#!/bin/ksh
#---------------------------------------------------------------------------
# Filename:    start_lcappmon
# Path:        /usr/es/sbin/cluster/local
# Node(s):     HACMP-Server; Hot-Standby LiveCache Server
# Info:        Startscript for HACMP-Application Monitoring
#---------------------------------------------------------------------------
# First set the environment and dump a timestamp
#---------------------------------------------------------------------------

# set logging mode
[[ "$VERBOSE_LOGGING" = high ]] && set -x

. /usr/es/sbin/cluster/local/hacmpr3.profile

#echo "start execution of $TOOLSDIR/start_lcappmon" >> $OUT
#echo `date` >> $OUT

#---------------------------------------------------------------------------
# Choice one (our the preferred one):
#   - get status of LiveCache at service/master node
#     Remark: Status must be twice the same to be reliable
#   - if status of LiveCache is not equal within two successive checks
#     -> do not trust the status and continue checking until it's twice the same
#     -> check two times and leave if it's not equal twice
#        meaning wait for next check - exit with returncode 0
#   - if status of LiveCache is "online" within two successive checks
#     -> it's all right - exit with returncode 0
#   - if status of LiveCache is "starting" within two successive checks
#     -> MaxDB starts up; wait until it's active
#     => while HACMP calls this script permanently - exit with returncode 0
#   - if status of LiveCache is "admin" within two successive checks
#     -> MaxDB is rebuilding; wait until it's no longer in admin state
#     => while HACMP calls this script permanently - exit with returncode 0
#   - if status of LiveCache is "standby" within two successive checks
#     -> Master was taken over and needs time to recover; just wait
#     => while HACMP calls this script permanently - exit with returncode 0
#   - else status of LiveCache is anything else within two successive checks
#     -> something seems to be going wrong - exit with returncode 12
# Choice two (maybe you like that even more),
# can also be done with standard HACMP Process Monitoring:
#   - check the DB-kernel processes (they are two)
#   - if they are both available
#     -> it's all right - exit with returncode 0
#   - else
#     -> something seems to be going wrong - exit with returncode 12
#---------------------------------------------------------------------------

# Choice one (our preferred one):

LCSTATE_1=$($LCBIN/dbmcli -n $LCSERVER_IP -d $LC_name -u control,control db_state | grep -v OK
| grep -v State)
sleep 5
LCSTATE_2=$($LCBIN/dbmcli -n $LCSERVER_IP -d $LC_name -u control,control db_state | grep -v OK
| grep -v State)

if [ $LCSTATE_1 = $LCSTATE_2 ]
then
   LCSTATE=$LCSTATE_2
else
   sleep 10
   LCSTATE_3=$($LCBIN/dbmcli -n $LCSERVER_IP -d $LC_name -u control,control db_state | grep -v
OK | grep -v State)
   if [ $LCSTATE_2 = $LCSTATE_3 ]
   then
      LCSTATE=$LCSTATE_3
   else
      exit 0
   fi
fi

case $LCSTATE in
   ONLINE)
#      echo "Master is ONLINE - it is working well" >> $OUT
      exit 0;;
   STARTING)
      echo "Master is STARTING up - be patient and wait until it's active" >> $OUT
      exit 0;;
   ADMIN)
      echo "Master is rebuilding, it's in ADMIN state - wait until it's ready" >> $OUT
      exit 0;;
   STANDBY)
      echo "Master is recovering, it's in STANDBY state - HACMP does nothing, just waits" >>
$OUT
      echo "contact the DB-Admin to check if all works well !!!" >> $OUT
      exit 0;;
   *)
      echo "state $LCSTATE of Master-LC not valid - something is going wrong" >> $OUT
      echo "=> let HACMP do a Takeover" >> $OUT
      exit 12;;
esac

# Choice two (easier but not prefered so it's commented out):

#if [ $(ps -ef | grep -v grep | grep kernel | grep $LC_name | wc -l) = 2 ]
#then
#   echo "Master processes are available - seems it is working well" >> $OUT
#   exit 0
#else
#   echo "state $LCSTATE of Master-LC not valid - something is going wrong" >> $OUT
#   echo "=> let HACMP do a Takeover" >> $OUT
#   exit 12
#fi

exit 0
```

9.3 HACMP Admin-Scripts and Commands

- Start HACMP ⇔ **HA_start_HACMP**

```
/usr/es/sbin/cluster/etc/rc.cluster -boot –I
```

- Stop HACMP ⇔ **HA_stop_HACMP**

```
/usr/es/sbin/cluster/utilities/clstop -N -s –g
```

- Stop HACMP with Takeover ⇔ **HA_stop_HACMP_takeover**

```
/usr/es/sbin/cluster/utilities/clstop -N -s –gr
```

- Stop Application Monitoring ⇔ **HA_stop_Appl_Monitoring**

```
/usr/es/sbin/cluster/events/utils/cl_RMupdate suspend_appmon LCMasterAppl
```

- Start Application Monitoring ⇔ **HA_start_Appl_Monitoring**

```
/usr/es/sbin/cluster/events/utils/cl_RMupdate resume_appmon LCMasterAppl
```

- Move a Resource Group to the takeover node ⇔ **HA_move_RG_to_remotenode**

```
#!/bin/ksh

if [ `hostname` = enhot1 ]
then
   REMOTE_NODE=node2
elif [ `hostname` = enhot2 ]
then
   REMOTE_NODE=node1
fi

/usr/es/sbin/cluster/utilities/clRGmove -g RG_LCmaster -m -n $REMOTE_NODE -p -i
```

- Move a Resource Group to the primary node ⇔ **HA_move_RG**

```
/usr/es/sbin/cluster/utilities/clRGmove -g RG_LCmaster –m –n Restore_Node_Priority_Order -p -i
```

- Check Status ⇔ **HA_check_status**
    - ➢ HACMP Resourcegroup handling and ownership
    - ➢ Livecache/DB
    - ➢ HACMP Node Order

```
#!/bin/ksh

while [ 1 ]
do
/usr/es/sbin/cluster/utilities/clRGinfo
/sapdb/programs/bin/dbmcli -d HOT -u control,control db_state
cat /usr/es/sbin/cluster/etc/clpol
sleep 10
done
```

10 List of references

[1] *SAPDB Hot Standby*, Jörg Mensing (SAP Labs Berlin, 20.Feb.2003)

[2] *esscli Command Reference and User's Guide*, Glenn Williamson, Amy Therrien, John Paveza & Stefan Jaquet (IBM Cooperation, 2002).

[3] *Command-Line Interfaces User's Guide*, - (IBM Cooperation,2002).

11 Notices and Disclaimers

For additional white papers on IBM storage solutions, go to:

- IBM TotalStorage Enterprise storage server white papers
  http://www.storage.ibm.com/disk/ess/whitepaper.htm
- IBM TotalStorage services white papers. Listing of whitepapers for different IBM Storage Services White Papers
  http://ssdweb01.storage.ibm.com/services/whitepapers/index.html
- Find white papers for IBM TotalStorage Virtualization Family. IBM TotalStorage Virtualization Family efficiently manage every aspect of your storage resources with IBM Storage Software: DFSMS, CICS, Data Path Optimizer, SAN Software, and Tivoli Software. IBM Storage Software is easy to use and implement, featuring open standards to ensure cross-platform compatibility, high-performance scalability. and robustness.
  http://www.storage.ibm.com/software/center/wpapers.html

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This information could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the product(s) and/or programs(s) at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectually property rights, may be used instead. It is the user's responsibility to evaluate and verify the operation of any non-IBM product, program or service.

THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR INFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted according to the terms and conditions of the agreements (*e.g.*, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. IBM is not responsible for the performance or interoperability of any non-IBM products discussed herein.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Trademarks
IBM®, the IBM logo, Enterprise Storage Server®, e-business on demand, and TotalStorage®, flashCopy are trademarks of International Business Machines Corporation in the United States, other countries, or both.

SAP, the SAP logo and all other SAP products and services mentioned herein are trademarks or registered trademarks of SAP AG in Germany and several other countries.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product or service names may be trademarks or service marks of others.